University of Alberta

EFFICIENT STOPPING RULES

by

Volodymyr Mnih

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Fall 2008

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-47368-9
Our file Notre référence
ISBN: 978-0-494-47368-9

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

The importance of sampling methods in machine learning is growing due to an ever-increasing number of datasets containing millions of records of biological, medical, or other types of data. Such datasets are often beyond the reach of many standard machine learning techniques because of high computational or space complexity of the algorithms. When making even a single pass through the data is prohibitive, sampling may offer a good solution. However, whenever sampling is employed, it is necessary to determine when to stop sampling in a principled manner. Taking too few samples may result in an algorithm that is not theoretically sound, while taking too many may waste valuable computational resources. We use the problem of estimating the mean of a bounded random variable up to a given relative error to show how the recently introduced empirical Bernstein bounds can be used to develop efficient stopping rules. We propose several new stopping rules, prove bounds on their expected stopping times, and demonstrate experimentally that the new rules can stop much earlier than the best competitors while offering the same probabilistic guarantees.

Acknowledgements

I would like to thank my supervisor Csaba Szepesvári for his excellent guidance. I am grateful for all the time you have put into helping me become a better researcher. I would also like to thank my brother Andriy for introducing me to the field of machine learning and for the constant encouragement. Thanks also goes to everyone in the RLAI lab for creating a great work environment. Finally, I would like to thank Anita and my family for their support.

Table of Contents

1 Introduction **1**

2 Related Work **4**
2.1 Algorithm $\mathcal{AA}$ 4
2.2 Nonmonotonic Adaptive Sampling 7
2.3 Asymptotic Approaches 8

3 Empirical Bernstein Stopping **12**
3.1 General Approach 12
3.2 The EBStop Algorithm 13
3.2.1 Stopping criterion 13
3.2.2 Choosing d_t 15
3.3 Analysis of EBStop 16
3.4 Effect of Range 21
3.4.1 The reduction approach 21
3.4.2 Upper bounds 22
3.4.3 Lower bound 22

4 Batch Sampling **24**
4.1 Batch Sampling 24
4.2 Mid-interval Stopping 26
4.3 Analysis of Batch Sampling 28

5 Experimental Results **31**
5.1 Experimental Setup 31
5.2 Effect of Variance 31
5.3 General Efficiency 34
5.3.1 Low Variance 34
5.3.2 High Variance 35
5.4 Coverage 36

6 Absolute Error **38**
6.1 Non-adaptive approach 38
6.2 Empirical Bernstein Stopping for Absolute Error 39
6.2.1 The Algorithm 39
6.2.2 Analysis 40
6.2.3 Mixture of Stopping Rules 42
6.3 Experimental Results 42
6.4 Conclusions 44

7 Conclusion **46**
7.1 Summary of Contributions 46
7.2 Future Work 46

8 Appendix **48**
8.1 Probability Inequalities 48
8.1.1 Hoeffding's Inequality 48
8.1.2 Empirical Bernstein Bounds 48

List of Tables

2.1 Probability of MCA failing for different values of t_{min}, $\epsilon = 0.1$, and $\delta = 0.1$. 10

4.1 Failure probability used to evaluate the stopping criterion after t samples by each algorithm. 28

List of Figures

5.1 Average number of samples required to find $(0.01, 0.1)$-approximations of $U(0, 1, m)$ random variables for $m = 1, 5, 10, 50, 100, 1000$. The results are averaged over 100 runs. 32

5.2 Average number of samples required to find $(0.1, 0.1)$-approximations of $U(a, b, 10)$ random variables with varying means. The results are averaged over 100 runs. 34

5.3 Average number of samples required to find $(0.1, 0.1)$-approximations of Bernoulli random variables with varying means. The results are averaged over 100 runs. 36

6.1 Comparison of absolute (ϵ, δ)-stopping rules on averages of m Uniform(0,1) random variables for $m = 1, 5, 10, 50, 100, 1000$. 43

6.2 Comparison of absolute (ϵ, δ)-stopping rules on a $U(0, 1, 3)$ random variable for different values of ϵ. 44

Chapter 1

Introduction

Consider the problem of deciding which of two poker players is better and by how much. It would not be unreasonable to define the better player as the one that would on average win more money per hand if the two players were to play an infinite number of hands. Since we cannot make the players play an infinite number of hands, an obvious approach is to make them play some finite number and declare the one who has won more money as the better player.

The problem becomes one of deciding how many hands need to be played. Clearly we want this number to be as small as possible. However, the fewer hands are played, the higher the probability that the weaker player wins more money through pure luck. These two competing objectives can be balanced by requiring that the number of hands to be played is as small as possible to guarantee that the wrong player is declared as being stronger with probability not exceeding some small threshold.

To define the problem more precisely, let $X_1, X_2, X_3, \ldots$ be independent, identically distributed (*iid*) random variables with mean μ. We will refer to an algorithm as a stopping rule if at time t it observes X_t and based on past observations decides whether to stop or continue sampling. If a stopping rule $\mathcal{S}$ returns $\hat{\mu}$ that satisfies

$$\mathbb{P}\left[|\hat{\mu} - \mu| \leq \epsilon|\mu|\right] \geq 1 - \delta, \tag{1.1}$$

then $\mathcal{S}$ is a (ϵ, δ)-stopping rule and $\hat{\mu}$ is an (ϵ, δ)-approximation of μ. If we let X_i be the random payoff for the first player for the i^{th} hand, we recall that, by our earlier definition, the first player is better if μ is greater than 0, the second player is better when μ is less than 0, and the magnitude of μ is the margin by which one of the players is stronger. By choosing $\epsilon = 1/2$, given any $0 < \delta < 1$, if $\hat{\mu}$ is an (ϵ, δ)-approximation to μ then $\hat{\mu}$ and μ will assume the same sign. Hence, an (ϵ, δ)-stopping rule can be used to solve the poker

problem.[1]

In general, estimating the expected value of a random variable through sampling, or Monte Carlo estimation, is a fundamental tool in many areas of science. In a clinical trial, one may be interested in estimating the probability that a new treatment succeeds, which can be seen as the expected value of an indicator random variable. A mathematician may be interested in approximating the permanent of a $0-1$ matrix through sampling because exact computation of this quantity is NP-hard.

In machine learning, the importance of sampling methods is growing due to an ever increasing number of datasets containing millions of records of biological, medical, or other types of data. Such datasets are often beyond the reach of many standard machine learning techniques because of poor computational or space complexity of the algorithms. In these cases, when even a single pass through a dataset can be prohibitive, sampling has emerged as a promising tool for scaling up machine learning algorithms [3, 8, 11].

As in the poker problem, whenever sampling is employed, a way of determining when enough samples have been taken is necessary, leading to the above described *stopping problem*. Taking too few samples may lead to a high-variance unreliable estimate. Taking too many samples, on the other hand, will produce an accurate estimate, but may be costly in terms of computational or laboratory resources.

Motivated by the above examples, this thesis examines the problem of finding an efficient (ϵ, δ)-stopping rule for bounded random variables. We consider the case of bounded random variables because it is possible to use finite sample tail bounds to obtain stopping rules with strict probabilistic guarantees for this setting. Although it would be possible to extend the results to the unbounded case when the random variables satisfy certain moment conditions (e.g., if they are sub-Gaussian) for the sake of simplicity we will not deal with this case here. We use the recently-introduced empirical Bernstein bounds to develop a new algorithm, EBGStop, that requires on the order of

$$\max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon|\mu|}\right)\left(\log\log\frac{R}{\epsilon|\mu|} + \log\frac{3}{\delta}\right) \tag{1.2}$$

samples to find an (ϵ, δ)-approximation of a random variable with range R (Theorem 4). Since, as it follows from a lower bound by Dagum et al. [4], any algorithm must take at least on the order of

$$\max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon|\mu|}\right)\cdot\log\frac{2}{\delta}, \tag{1.3}$$

[1] In the poker problem assuming that the payoffs are *iid* rules out players who adapt their strategy between games. Poker programs will often meet this condition.

samples (cf. Theorems 1 & 3), EBGStop is close to achieving the optimal bound. We also show that EBGStop often stops much earlier than the best known stopping rule for bounded random variables in practice. Most of our results on (ϵ, δ)-approximations have appeared in [12], but the treatment provided by this thesis is more complete.

We then apply our techniques to the problem of estimating the mean of a bounded random variable up to ϵ *absolute* error with probability at least $1 - \delta$. We present a simple algorithm that requires on the order of

$$\max\left(\frac{\sigma^2}{\epsilon^2}, \frac{R}{\epsilon}\right)\left[\log\log\frac{R}{\epsilon} + \log\frac{3}{\delta}\right] \tag{1.4}$$

samples. While our algorithm often requires much fewer samples than the standard approach of taking

$$\frac{R^2}{2\epsilon^2} \cdot \log\frac{2}{\delta} \tag{1.5}$$

samples, our approach often stops later when the variance is large. We then introduce a stopping rule that uses a mixture of two stopping rules and show that it often stops much earlier than the standard approach while never exceeding its stopping time by more than a constant.

Chapter 2

Related Work

In this chapter, we present the relevant work on (ϵ, δ)-stopping rules. We start by examining sound (ϵ, δ)-stopping rules and then consider some approximate approaches based on the central limit theorem.

2.1 Algorithm $\mathcal{AA}$

Dagum et al. [4] present an algorithm for finding an (ϵ, δ)-approximation of the mean of a random variable distributed in $[0, 1]$. Their approximation algorithm, referred to as $\mathcal{AA}$ for short, is optimal in the sense that the expected number of samples it takes is within a universal multiplicative constant of any other algorithm for finding an (ϵ, δ)-approximation.

The next theorem proved by Dagum et al. [4] about (ϵ, δ)-approximations is the key to understanding how $\mathcal{AA}$ works. But before the theorem let us introduce formally introduce the concept of universal (ϵ, δ)-stopping rules:

Definition 1. *Consider a stopping rule $\mathcal{S}$. Let a distribution D be supported on $[0, 1]$, $\mu > 0$ be its expected value, $\hat{\mu}_{(\epsilon,\delta)}$ be the approximation to μ returned by $\mathcal{S}$ when run with parameters (ϵ, δ) on* iid *samples drawn from the distribution D, and let $N_{(\epsilon,\delta)}$ be the time when the rule stops. If for any such distribution D and any $(\epsilon, \delta) \in (0, 1)^2$ it holds that*

1. $\mathbb{E}[N_{(\epsilon,\delta)}] < \infty$, *and*
2. $\mathbb{P}\left[\mu(1-\epsilon) \leq \hat{\mu}_{(\epsilon,\delta)} \leq \mu(1+\epsilon)\right] > 1 - \delta$,

then $\mathcal{S}$ is called a universal (ϵ, δ)*-stopping rule.*

Theorem 1. *Let $\mathcal{S}$ be a universal (ϵ, δ)-stopping rule. Pick any $(\epsilon, \delta) \in (0, 1)^2$ and any distribution D supported on $[0, 1]$ whose mean is positive. Let $N_{(\epsilon,\delta)}$ be the time when $\mathcal{S}$*

stops on this problem with parameters (ϵ, δ). *Then*

$$\mathbb{E}[N_{(\epsilon,\delta)}] \geq c \cdot \max(\sigma^2, \epsilon\mu) \cdot \frac{1}{\epsilon^2\mu^2} \log \frac{2}{\delta}, \tag{2.1}$$

where μ *is the mean of* D *and* σ^2 *is its variance, and* $c > 0$ *is a constant that is independent of* $\mathcal{S}$, D, ϵ, *and* δ.

Theorem 1 gives the minimum number of samples an algorithm needs to take on the average in order to always produce an (ϵ, δ)-approximation of μ. The $\mathcal{AA}$ algorithm can be seen as an attempt to reverse-engineer an optimal stopping rule through Theorem 1.

First, Dagum et al. found a constant c' that guarantees that if $n = c' \cdot \max(\sigma^2, \epsilon\mu) \cdot \frac{1}{\epsilon^2\mu^2} \log \frac{2}{\delta}$ and $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$, then

$$\mathbb{P}\left[|\hat{\mu} - \mu| \leq \epsilon\mu\right] \geq 1 - \delta.$$

If μ and σ^2 were known, one could compute n and simply average n samples to obtain an (ϵ, δ)-approximation of μ. However, μ is the quantity of interest in the first place, so Dagum et al. instead compute an upper bound on n using approximations of μ and σ^2 that are within a constant factor of the true values with high probability.

To obtain approximations of μ and σ^2 that are used to compute N, Dagum et al. use the Stopping Rule Algorithm (SR), pseudocode for which appears as Algorithm 1. Like $\mathcal{AA}$, given $\epsilon > 0$ and $\delta \in (0, 1)$ the SR algorithm returns an (ϵ, δ)-approximation of μ. However, the expected number of samples taken by SR is on the order of $\frac{1}{\epsilon^2\mu} \log \frac{2}{\delta}$, suggesting that there may be a more efficient algorithm that, in some cases, would take $1/\epsilon$ times fewer samples.

Algorithm 1 Stopping Rule Algorithm

```
t ← 0
S ← 0
Υ ← 4(e − 2) log(2/δ)/ε²
Υ₁ ← 1 + (1 + ε)Υ
while S ≤ Υ₁ do
  t ← t + 1
  Obtain X_t
  S ← S + X_t
end while
return Υ₁/t
```

Pseudocode for the $\mathcal{AA}$ algorithm is presented as Algorithm 2, where, for clarity, $X_1, X_2, \ldots$ and $X'_1, X'_2, \ldots$ denote two groups of *iid* random variables distributed with

mean μ and σ^2. There are three steps to the algorithm. In the first step, the SR algorithm is used to obtain a $(\min(1/2, \sqrt{\epsilon}), \delta/3)$-approximation of μ. In the second step, a high-probability estimate of σ^2 is found by using the estimate of μ to determine the necessary number of samples. Finally, the estimates of μ and σ^2 are combined into an estimate of $\max(\sigma^2, \epsilon\mu)$, which in turn is used to determine the number of samples necessary to obtain an (ϵ, δ)-approximation of μ. Note that the third step reuses the samples used in the first step before obtaining new ones.

Algorithm 2 Algorithm $\mathcal{AA}$

$\Upsilon_1 \leftarrow 2(1+\sqrt{\epsilon})(1+2\sqrt{\epsilon})(1+\log\frac{3}{2}/\log\frac{2}{\delta})\Upsilon$

/* Use the Stopping Rule Algorithm on $X_1, X_2, \ldots$ to find approximation of μ */
$\hat{\mu}' \leftarrow (\min(1/2, \sqrt{\epsilon}), \delta/2)$-approximation of μ

/* Find approximation of $\max(\sigma^2, \epsilon\mu)$ using $X'_1, X'_2, \ldots$
$N \leftarrow \Upsilon_1 \cdot \epsilon/\hat{\mu}'$
$S \leftarrow 0$
for $i = 1, \ldots, N$ **do**
  $S \leftarrow S + (X'_{2i-1} - X'_{2i})^2/2$
end for
$\rho \leftarrow \max(S/N, \epsilon\hat{\mu}')$

/* Find final approximation of μ using $X_1, X_2, \ldots$ */
$N \leftarrow \Upsilon_1 \cdot \rho/\hat{\mu}'^2$
$S \leftarrow 0$
for $i = 1, \ldots, N$ **do**
  $S \leftarrow S + X_i$
end for
$\hat{\mu} \leftarrow S/N$

return $\hat{\mu}$

Finally, Dagum et al. prove that for any random variable X distributed in $[0, 1]$, $\epsilon > 0$, and $\delta \in (0, 1)$, if $\hat{\mu}$ is the estimate produced by $\mathcal{AA}$ and N is the stopping time of $\mathcal{AA}$, then $\mathcal{AA}$ satisfies the conditions of Theorem 1 and there exists a universal constant c such that

$$\mathbb{E}[N] \leq c \cdot \max(\sigma^2, \epsilon\mu) \cdot \frac{1}{\mu^2\epsilon^2} \log\frac{2}{\delta}. \tag{2.2}$$

It seems that extending the $\mathcal{AA}$ algorithm to the more general setting of bounded random variables should be trivial, but this is not the case. The main technique used by the $\mathcal{AA}$ algorithm relies heavily on the fact that the sum of n samples from a nonnegative random variable is non-decreasing as a function of n. This is not true for a sum of n bounded

random variables, hence $\mathcal{AA}$ cannot be extended to this case. Nevertheless, the results of Dagum et al. provide important insights into our problem.

2.2 Nonmonotonic Adaptive Sampling

Domingo et al. [6] propose the Nonmonotonic Adaptive Sampling (NAS) algorithm for finding an (ϵ, δ)-approximation of the mean of a bounded random variable. Pseudocode for the NAS algorithm is shown as Algorithm 3.

Algorithm 3 Algorithm NAS

```
α ← ∞
u ← 0
t ← 0
while |u| < α(1 + 1/ε) do
  t ← t + 1
  Obtain X_t
  u ← X̄_t
  α ← sqrt((1/2n) log(n(n + 1)/δ))
end while
return X̄_t
```

The idea behind the NAS algorithm is simple. After observing t samples, a $1 - d_t$ confidence interval for μ, where $d_t = \delta/(t(t+1))$, is constructed around $\overline{X}_t$ using Hoeffding's inequality. Setting α to be half the width of this confidence interval, the algorithm terminates if $|\overline{X}_t| < \alpha(1 + 1/\epsilon)$ and returns $\overline{X}_t$. To see why $\overline{X}_t$ is an (ϵ, δ) approximation when NAS terminates, suppose that the NAS algorithm stopped after t samples and that all confidence intervals contain μ. It follows that

$$|\overline{X}_t - \mu| \leq \alpha \leq \epsilon(|\overline{X}_t| - \alpha) \leq \epsilon|\mu|. \tag{2.3}$$

The first and third inequalities follow from the fact that all the confidence intervals hold, while the second inequality can be obtained by rearranging the stopping condition. Finally, it follows by the union bound that Equation (2.3) holds with probability at least $1 - \delta$ since

$$\sum_{t=1}^{\infty} \frac{\delta}{t(t+1)} \leq \delta. \tag{2.4}$$

Hence, upon termination of the NAS algorithm, $\overline{X}_t$ is an (ϵ, δ)-approximation of μ.

Domingo et al. argue that given any $\epsilon > 0$, $\delta > 0$, and if $X_1, X_2, \ldots$ are *iid* bounded random variables with mean $\mu \neq 0$, then there exists a universal constant c such that

$$\mathbb{E}[N] \leq c \cdot \frac{1}{\mu^2 \epsilon^2} \cdot \left(\log \frac{1}{\epsilon|\mu|} + \log \frac{2}{\delta} \right). \tag{2.5}$$

Unlike the equivalent bound for the $\mathcal{AA}$ algorithm, Equation (2.5) contains an additional $\log(1/\epsilon|\mu|)$ term. This term comes from the use of a union bound. Domingo et al. also show that it can be reduced to $\log\log(1/\epsilon|\mu|)$ through the use of "geometric sampling".[1] Concentrating on non-negative valued random variables, it is also interesting to note that the bound for the NAS algorithm does not contain the $\max(\sigma^2, \epsilon\mu)$ term that is present in Equation (2.2), suggesting that NAS will perform poorly when $\max(\sigma^2, \epsilon\mu) \ll 1$.

2.3 Asymptotic Approaches

The $\mathcal{AA}$ and NAS algorithms rely, directly or indirectly, on finite sample tail bounds, such as Hoeffding's inequality. An alternative approach is to use deviation bounds based on the Central Limit Theorem [15]. While such an approach can only offer asymptotic guarantees, it can result in earlier stopping times. In this section, we discuss several asymptotic approaches to finding (ϵ, δ)-approximations and provide some insight into how they can fail.

Let $X_1, X_2, \ldots$ are *iid* random variables with finite mean μ and finite variance $\sigma^2 > 0$, and let $\overline{\Phi} = 1 - \Phi$, where Φ denotes the standard normal cumulative density function. Let $\overline{X}_t$ be the average of $X_1, X_2, \ldots, X_t$, V_t be the empirical variance:

$$\begin{aligned} \overline{X}_t &= \frac{1}{t}\sum_{s=1}^{t} X_s, \\ V_t &= \frac{1}{t}\sum_{s=1}^{t}(X_s - \overline{X}_t)^2. \end{aligned} \tag{2.6}$$

Then V_t converges to σ^2 in distribution and hence according to the Central Limit Theorem (Theorem 1.13 in [5]) and Slutsky's Theorem (Theorem 1.5 in [5]),

$$\mathbb{P}\left[\frac{\sqrt{t}(\overline{X}_t - \mu)}{\sqrt{V_t}} > u\right] \to \overline{\Phi}(u)$$

If we define

$$c_t = \frac{\overline{\Phi}^{-1}(\delta/2)\sqrt{V_t}}{\sqrt{t}}, \tag{2.7}$$

then $\lim_{t\to\infty} \mathbb{P}\left[|\overline{X}_t - \mu| \le c_t\right] = 1 - \delta$, hence, in the limit, $(\overline{X}_t - c_t, \overline{X}_t + c_t)$ is a $1 - \delta$ confidence interval for μ. Such approximate confidence intervals are generally much tighter than confidence intervals obtained from Hoeffding's inequality or the empirical Bernstein bound.

[1] Geometric sampling will be explained in Chapter 4.

Domingo et al. analyze a version of the NAS algorithm that uses CLT-based confidence intervals [6]. They argue that the expected number of samples required by this variant of NAS still scales with $1/\epsilon^2\mu^2$, but the constants are significantly reduced, resulting in earlier stopping times. However, Domingo et al. do not consider the effect of the variance in their analysis, which suggests that it may be possible to prove a tighter bound.

A similar approach was taken by Holmes et al., who developed an asymptotic (ϵ, δ)-stopping rule for the purpose of approximating intractable statistical summations [11]. Their Monte Carlo approximation algorithm, which we will refer to as MCA, is shown as Algorithm 4. The MCA algorithm is representative of asymptotic approaches to stopping in that it makes use of CLT-based confidence intervals and it does not make use of a union bound [14].

Algorithm 4 Algorithm MCA

```
t ← 0
t_needed ← t_min
while t < t_needed do
  while t < t_needed do
    t ← t + 1
    Obtain X_t
  end while
  t_needed ← z²_{δ/2} (1+ε)²/ε² · V_t/X̄_t²
end while
return X̄_t
```

Holmes et al. derive MCA from the observation that if c is half the width of a $1-\delta$ confidence interval for μ as defined by Equation 2.7, then $\hat{\mu}$ is an (ϵ, δ)-approximation of μ whenever $c \leq \epsilon(\hat{\mu} - c)$. This is in fact the stopping condition used by the CLT-based NAS algorithm, but unlike the NAS algorithm, MCA does not check the stopping condition after each sample. Instead, the MCA algorithm begins by taking some predetermined number of samples t_{min} before checking the stopping condition for the first time. Holmes et al. observe that if the stopping condition is not satisfied, one can rewrite it as

$$t \geq z_{\delta/2}^2 \frac{(1+\epsilon)^2}{\epsilon^2} \cdot \frac{V_t}{\overline{X}_t^2}, \tag{2.8}$$

where z_δ is a $1-\delta$ quantile of a standard normal distribution, to obtain a lower bound on the number of required samples. In MCA, Equation 2.8 is used to determine when the stopping condition should be checked next if it is not already satisfied.

The MCA algorithm is closely related to Stein's two-stage method for finding fixed width confidence intervals (see Chapter 13 of [13]), a problem we will consider in Chap-

t_{min}	30	100	500
$Laplace(0.1, 1)$	0.355	0.222	0.131
$Gaussian(0.1, 1)$	0.251	0.159	0.103

Table 2.1: Probability of MCA failing for different values of t_{min}, $\epsilon = 0.1$, and $\delta = 0.1$.

ter 6. Given a sequence of *iid* random variables $X_1, X_2, \ldots$ with mean μ and variance σ^2, both unknown, the goal is to find a confidence interval of width 2ϵ that contains μ with probability at least $1 - \delta$. Stein's two-stage procedure begins by taking some fixed number m samples in the first stage. Using these samples, a stopping time T is computed as

$$T = \max\left(m, t^2_{(m-1,\delta/2)}\frac{V_m}{\epsilon^2} + 1)\right),$$

$t_{(m,\delta)}$ is the $1 - \delta$ quantile of Student's t-distribution with m degrees of freedom. In the second stage, Stein's procedure takes a further $T - m$ samples. When X_i are normally distributed, this procedure has been shown to take roughly twice as many samples as a stopping rule that knows the true variance. Nevertheless, the rule gives the desired coverage in this case.

One important question is what effect does the use of approximate confidence intervals have on the properties of such two-stage procedures? If statistical folklore is to be believed, setting t_{min} to 30 should ensure that the normal approximation holds. Hence, the probability that MCA produces an approximation with relative error less than ϵ, also known as the *coverage* of a stopping rule, should not be smaller than $1 - \delta$.

We explored the validity of this claim by estimating the coverage of MCA for different values of t_{min} in two different scenarios. If p is the probability that MCA produces an estimate with relative error greater than ϵ for some random variable, then $p = \mathbb{E}[\mathbb{I}\{|\mu - \hat{\mu}| \geq \epsilon\mu\}]$, where $\mathbb{I}\{\mathcal{A}\}$ denotes the indicator random variable for event $\mathcal{A}$. We will refer to p as the *failure probability* of a stopping rule. Since we are primarily interested in determining whether $p < \delta$ or $p > \delta$, we used a stopping rule to find a $(0.1, 0.1)$-approximation of the mean of the random variable $\mathbb{I}\{|\mu - \hat{\mu}| \geq \epsilon\mu\} - \delta$.

We estimated the failure probability of MCA on $Laplace(\mu = 0.1, b = 1)$ and $Normal(\mu = 0.1, \sigma^2 = 1)$ random variables for $t_{min} = 30, 100, 500$. The Laplace distribution has high kurtosis so we can expect MCA to fail with probability greater than δ on it. However, when sampling from a Normal distribution MCA should fail with probability close to δ since in that case the only approximate step is that the variance and the mean are both estimated based on data.

The results are shown as Table 2.3. It is clear that when t_{min} is too low, MCA can fail

with probability much larger than δ. In particular, the claim that CLT-based approximations are accurate when the number of samples is greater than 30 seems far from true. On both random variables, when t_{min} was set to 30 the probability of MCA failing was between 2.5 and 3.5 higher than the desired value of $\delta = 0.1$. While MCA was much closer to achieving the desired failure probability of 0.1 when t_{min} was set to 500, this is not a guarantee that this will be the case for other random variables. Some attempts to make two-stage procedures such as Stein's more robust have been made, for example by employing the bootstrap [9], however the resulting guarantees are still asymptotic.

While the earlier stopping times provided by asymptotic approaches are appealing, they should not be applied blindly because if they are not properly tuned they can significantly exceed the desired failure probability of δ. Note that there are two sources of the error: First, instead of using the true variance we use an empirical estimate. Second, the CLT is asymptotic. Since the convergence in the CLT is of order $\Theta(1/\sqrt{n})$ as it follows from Cramer's theorem (Theorem 13.1 in [5]), correcting for the error committed when using $\overline{\Phi}$ would introduce an intolerably large cost (the sample size would be $\Omega(1/\delta^2)$). In the rest of this thesis we will only consider stopping rules that can offer strict probabilistic guarantees, yet avoid this pitfall.

Chapter 3

Empirical Bernstein Stopping

In this chapter, we develop a near-optimal stopping rule for finding an (ϵ, δ)-approximation of the mean of a bounded random variable.

3.1 General Approach

We begin by describing the general approach taken in the design of our algorithm. To reiterate, the goal is to construct a stopping rule with the following two properties:

1. For any $\mu \neq 0$, the stopping rule should stop with probability one.
2. The estimate $\hat{\mu}$ returned by the stopping rule should satisfy

$$\mathbb{P}\left[|\hat{\mu} - \mu| \leq \epsilon|\mu|\right] \geq 1 - \delta$$

To gain some insight into how a stopping rule can be made to satisfy the second property, let $\mathcal{F}$ be the event that the stopping rule fails, i.e. returns an estimate $\hat{\mu}$ that is not an ϵ-approximation of μ, and let T be the random time at which the stopping rule terminates. By the law of total probability, it follows that

$$\mathbb{P}\left[\mathcal{F}\right] = \sum_{t=1}^{\infty} \mathbb{P}\left[\mathcal{F} \cap \{T = t\}\right]. \tag{3.1}$$

The key idea behind our stopping rule is to define a nonnegative sequence $\{d_t\}$ satisfying $\sum_{t=1}^{\infty} d_t \leq \delta$, and construct the stopping rule such that $\mathbb{P}\left[\mathcal{F} \cap \{T = t\}\right] \leq d_t$. To facilitate this we define a new sequence $\{c_t\}$ where c_t is half the width of a $1 - d_t$ confidence interval for μ given t samples. The stopping criterion is then constructed so that if it is satisfied after t samples and if the confidence interval for μ computed using the sample and c_t does not fail then the returned estimate $\hat{\mu}$ is an ϵ-approximation of μ. This ensures

that $\mathbb{P}[\mathcal{F} \cap \{T = t\}] \leq d_t$, and when combined with Equation (3.1) guarantees that the stopping rule will fail with probability at most δ.

The same general approach was followed by Domingo et al. [6] in the design of their NAS algorithm, but since we construct the sequence c_t using empirical Bernstein bounds (see below) instead of Hoeffding bounds, our stopping rules are able to take advantage of variance like the $\mathcal{AA}$ algorithm. However, our approach works with absolute values of the sample means $\overline{X}_t$, and, unlike the $\mathcal{AA}$ algorithm, our stopping rules do not require the samples to be almost surely nonnegative.

3.2 The EBStop Algorithm

In this section, we present the basic version of our stopping rule, EBStop.

3.2.1 Stopping criterion

First, let $d_t = c/t^p$ where $c = \delta \cdot (p-1)/p$ and $p > 1$. This merely ensures that $\sum_{t=1}^{\infty} d_t \leq \delta$, but we will discuss this particular choice of $\{d_t\}$ in Section 3.2.2. Also let c_t be half the width of a $1 - d_t$ confidence interval for μ as defined by the empirical Bernstein bound given t samples (see Section 8.1.2)

$$c_t = \sqrt{\frac{2V_t \log(3/d_t)}{t}} + \frac{3R \log(3/d_t)}{t}, \tag{3.2}$$

and define the event $\mathcal{E}$ as

$$\mathcal{E} = \bigcap_{t \geq 1} \{|\overline{X}_t - \mu| \leq c_t\}. \tag{3.3}$$

Here $\overline{X}_t$ is the same mean of the first t samples and V_t is the sample variance (cf. Equation (2.6)). By construction, event $\mathcal{E}$ holds with probability at least $1 - \delta$. We now construct a stopping criterion that is guaranteed to return an (ϵ, δ)-approximation of μ given that event $\mathcal{E}$ holds.

From Equation (3.3) we know that $|\overline{X}_t - \mu| \leq c_t$ for all $t \in \mathbb{N}^+$. Since a confidence interval for the absolute value of the mean of a random variable is no wider than the equivalent confidence interval for the mean itself, it follows that $||\overline{X}_t| - |\mu|| \leq c_t$, which implies $|\overline{X}_t| - c_t \leq |\mu|$. It is then easy to see that if

$$c_t \leq \epsilon(|\overline{X}_t| - c_t), \tag{3.4}$$

then

$$||\overline{X}_t| - |\mu|| \leq c_t \leq \epsilon(|\overline{X}_t| - c_t) \leq \epsilon|\mu|.$$

Hence, if we stop when Inequality (3.4) holds, $|\overline{X}_t|$ is within ϵ relative error of $|\mu|$. We rearrange Inequality (3.4) as

$$c_t \leq \frac{\epsilon}{1+\epsilon} |\overline{X}_t| \tag{3.5}$$

for convenience to obtain the stopping condition of our first (ϵ, δ)-stopping rule, EBStop-Simple. Pseudocode for EBStopSimple is shown as Algorithm 5.

Algorithm 5 Algorithm EBStopSimple

```
c_t ← ∞
t ← 0
Obtain X_1
while c_t > ε/(1+ε) |X̄_t| do
    t ← t + 1
    Obtain X_t
    Compute c_t according to (3.2)
end while
return X̄_t
```

While it can be shown that EBStopSimple comes close to achieving the lower bound of Dagum et al. from Theorem 1, we make two simple improvements to EBStopSimple before providing a theoretical analysis of stopping times.

First, we show that the $(1+\epsilon)$ term in Inequality (3.5) can be discarded. Let $l(t) = |\overline{X}_t| - c_t$ and $u(t) = |\overline{X}_t| + c_t$. We have seen that $\mathbb{P}\left[\cap_{t\geq 1}\{l(t) \leq |\mu| \leq u(t)\}\right] > 1-\delta$. Now, consider an algorithm that stops at the first time T when

$$(1+\epsilon)l(T) \geq (1-\epsilon)u(T) \tag{3.6}$$

and returns the estimate

$$\hat{\mu} = 1/2 \cdot \text{sgn}(\overline{X}_T)\left[(1+\epsilon)l(T) + (1-\epsilon)u(T)\right]. \tag{3.7}$$

It is easy to show that for our choice of $l(t)$ and $u(t)$, Inequality (3.6) is equivalent to $c_T \leq \epsilon|\overline{X}_T|$. To show that the estimate defined in Equation (3.7) is an (ϵ, δ)-approximation consider the event $\mathcal{E}$ when for any t, $\overline{X}_t - c_t \leq \mu \leq \overline{X}_t + c_t$. On this event,

$$\begin{aligned} |\hat{\mu}| &= 1/2 \cdot [(1+\epsilon)l(T) + (1-\epsilon)u(T)] \\ &\geq (1-\epsilon)u(T) \\ &\geq (1-\epsilon)|\mu|. \end{aligned}$$

Here the first inequality follows from the stopping condition (3.6) and the second follows

by the definition of $\mathcal{E}$. Similarly,

$$\begin{aligned} |\hat{\mu}| &= 1/2 \cdot [(1+\epsilon)l(T) + (1-\epsilon)u(T)] \\ &\leq (1+\epsilon)l(T) \\ &\leq (1+\epsilon)|\mu|. \end{aligned}$$

Further, since $c_T < |X_T|$, the signs of $\overline{X}_t$ and μ must agree on $\mathcal{E}$. Thus, on $\mathcal{E}$, $\hat{\mu}$ is an ϵ-approximation to μ. Since by construction $\mathbb{P}[\mathcal{E}] \geq 1-\delta$, we get that the stopping rule returns an ϵ-estimate with probability at least $1-\delta$.

The second improvement is based on the observation that when conditioning on event $\mathcal{E}$, one can use the smallest of the confidence intervals constructed at any time $1 \leq s \leq t$ as the confidence interval at time t instead of c_t. When c_t is constructed from the empirical Bernstein bound, this construction can result in tighter confidence intervals, which in turn lead to earlier stopping times. Based on this, we can refine our definitions of $l(t)$ and $u(t)$ by setting $l(t)$ to $\max_{s \leq t} \left(|\overline{X}_s| - c_s\right)$ and $u(t)$ to $\min_{s \leq t} \left(|\overline{X}_s| + c_s\right)$.

We incorporate the above improvements into EBStopSimple to obtain a new algorithm, EBStop. The pseudocode for EBStop is shown as Algorithm 6.

Algorithm 6 Algorithm EBStop

```
t ← 1
l(t) ← 0
u(t) ← ∞
Obtain X_1
while (1+ε)l(t) < (1−ε)u(t) do
  t ← t + 1
  Obtain X_t
  Compute c_t according to (3.2)
  l(t) ← max(l(t−1), |X̄_t| − c_t)
  u(t) ← min(u(t−1), |X̄_t| + c_t)
end while
return sgn(X̄_t) · 1/2 · [(1+ε)l(t) + (1−ε)u(t)]
```

3.2.2 Choosing d_t

While we have already stated that $\{d_t\}$ should be nonnegative and should sum to δ, another restriction on the sequence is necessary to guarantee that EBStop will stop with probability one. While the reason will become clear in the next section, $\{d_t\}$ must satisfy

$$\lim_{t \to \infty} \frac{\log(3/d_t)}{t} = 0.$$

Hence, d_t should not decay too quickly, or EBStop will never terminate. Our particular choice of $\{d_t\}$ satisfies all of the above criteria and is both efficient in practice and mathematically convenient. The exact form of this sequence is a parameter of our algorithm and offers a way of incorporating prior knowledge.

3.3 Analysis of EBStop

In this section we prove that EBStop is an (ϵ, δ)-stopping rule and provide an analysis of its stopping times. We show that EBStop comes close to achieving the theoretical lower bound given in Theorem 1. We begin by stating a key technical result due to Audibert et al. [1].

Lemma 1. *Let U be a real-valued random variable such that almost surely $U \leq b$ for some $b \in \mathbb{R}$. Let $b' = b - \mathbb{E}[U]$, and $b_+ = \max(b, 0)$. Let $U_1, \ldots, U_n$ be i.i.d. copies of U and $\overline{U}_t = 1/t \sum_{s=1}^{t} U_s$. Then for any $x > 0$ the followings hold:*

- *with probability at least $1 - e^{-x}$, simultaneously for $1 \leq i \leq t$,*

$$i(\overline{U}_i - \mathbb{E}[U]) \leq \sqrt{2t\mathbb{E}[U^2]x} + b_+ x/3; \tag{3.8}$$

- *with probability at least $1 - e^{-x}$, simultaneously for $1 \leq i \leq t$,*

$$i(\overline{U}_i - \mathbb{E}[U]) \leq \sqrt{2t\mathbb{V}[U]x} + b' x/3. \tag{3.9}$$

Proof. See [1]. □

Lemma 1 can be used to derive a high probability upper bound on the sample variance, which is needed in order to show that the expected number of samples taken by EBStop depends on the true variance.

Lemma 2. *Let $X_1, \ldots, X_t$ be* iid *random variables such that for all $1 \leq i \leq t$, almost surely $0 \leq X_i \leq 1$. Let $V_t = \frac{1}{t} \sum_{i=1}^{t} (X_i - \overline{X}_t)^2$. Then, for any $x > 0$, with probability at least $1 - 3e^{-x}$,*

$$V_t \leq \sigma^2 + \sqrt{2\sigma^2 x/t} + x/3t. \tag{3.10}$$

Proof. The application of Inequality (3.8) with the choice $U_j = (X_j - \mathbb{E}[X_1])^2$, $i = t$, yields that with probability at least $1 - e^{-x}$,

$$\overline{U}_t \leq \sigma^2 + \sqrt{2\mathcal{V}x/t} + x/3t, \tag{3.11}$$

where $\mathcal{V} \triangleq \mathbb{E}[(X_1 - \mathbb{E}[X_1])^4]$. Now, $\overline{U}_t = V_t + (\overline{X}_t - \mu)^2 \geq V_t$, hence from (3.11) it also follows that

$$V_t \leq \sigma^2 + \sqrt{2\mathcal{V}x/t} + x/3t.$$

Using $\mathcal{V} \leq \sigma^2$, which holds since $X_i \in [0,1]$, we arrive at the desired result. □

Before proceeding to the main result, we prove a technical lemma that provides an upper bound on the solution of a type of equation that arises in the analysis of stopping times.

Lemma 3. *Let a, k be positive real numbers. If t' is a solution to*

$$\frac{\log at}{t} = k, \tag{3.12}$$

then

$$t' \leq \frac{2}{k} \log \frac{2a}{k}. \tag{3.13}$$

Further, if t' is as above and $t \geq t'$ then $\log(at)/t \leq k$.

Proof. The solution of Equation (3.12) can be seen as the intersection point between a line and a logarithmic curve when we rewrite the equation as

$$\log at = kt. \tag{3.14}$$

First, note that at $t = 1/k$, the slope of the line equals the slope of the tangent to the logarithmic curve. Because log is concave, for $t^0 > 1/k$ the intersection of the line tangent to $\log at$ at t^0 with the line kt is an upper bound on t'. Substituting $\log t^0 + 1/t^0 \cdot (t - t^0)$ with the choice of $t^0 = 2/k$ for $\log at$ in Equation (3.14) and solving for t yields

$$t = \frac{2}{k}\left[\log \frac{2a}{k} - 1\right]. \tag{3.15}$$

We obtain the Lemma by dropping the $-2/k$ term for convenience. □

Finally, we present a theorem that summarizes the main properties of EBStop. In order to simplify the analysis, we restrict it to the case of random variables with range $[0,1]$.

Theorem 2. *Let X be a random variable distributed with range $[0,1]$. Let $\mu = \mathbb{E}[X]$ and $\sigma^2 = \mathbb{V}[X]$ and assume $\mu \neq 0$. Let T be the stopping time of Algorithm EBStop on X, where c_t is defined by Equation (3.2) with $d_t = \delta(p-1)/(pt^p)$, where $p > 1$. Then the following properties hold:*

1. *There exists a constant $C = C_p$ such that for any $0 < \delta < 1/2$,*

$$\mathbb{P}\left[T > C \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{1}{\epsilon|\mu|}\right)\left(\log \frac{1}{\epsilon|\mu|} + \log \frac{2}{\delta}\right)\right] \leq 2\delta.$$

2. *If $p > 2$, there exists a constant $C' = C'_p$ such that*

$$\mathbb{E}[T] < C' \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{1}{\epsilon|\mu|}\right)\left(\log\frac{1}{\epsilon|\mu|} + \log\frac{2}{\delta}\right).$$

3. *The estimate $\hat{\mu}$ as returned by the EBStop algorithm is an (ϵ, δ)-approximation of μ.*

Proof of Part 1. When Algorithm EBStop stops at time T, the stopping condition implies that

$$\begin{aligned}&(1+\epsilon)\max_{s\le T}\left(|\overline{X}_s| - c_s\right) \ge (1-\epsilon)\min_{s\le T}\left(|\overline{X}_s| + c_s\right), \quad \text{and}\\ &|\overline{X}_T| \ge c_T.\end{aligned} \tag{3.16}$$

Since analyzing the stopping criterion directly is cumbersome, we will state a sequence of stopping conditions, each more conservative and easier to analyze than the previous until we arrive at a condition that can be solved for the stopping time. First, consider dropping the max and min from the first half of Inequality (3.16) and rearranging the terms, resulting in

$$\epsilon|\overline{X}_T| \ge c_T. \tag{3.17}$$

Since Inequality (3.16) holds only when Inequality (3.17) holds, it suffices to upper bound the stopping time of algorithm EBStop with Inequality (3.17) as its stopping criterion.Thus if we redefine T to be the first time when (3.17) holds then it suffices to upper bound T.

Now, consider the event $\mathcal{E}$ when none of the confidence intervals fail:

$$\mathcal{E} = \bigcap_{t\ge 1}\{|\overline{X}_t - \mu| \le c_t\}. \tag{3.18}$$

In what follows, unless told otherwise, we will always assume that this event holds. Since, on $\mathcal{E}$, $|\overline{X}_t| \ge |\mu| - c_t$ holds for any t, if T' is the first time when $\epsilon(|\mu| - c_{T'}) \ge c_{T'}$ holds then $T \le T'$. Redefining T to be T', our aim now is to bound T'. The new stopping criterion then becomes

$$\frac{\epsilon|\mu|}{1+\epsilon} \ge c_T = \sqrt{\frac{2V_T\log(1/\delta_T)}{T}} + \frac{3\log(1/\delta_T)}{T}, \tag{3.19}$$

where we used the definition of c_t (cf. Equation (3.2)) and we define $1/\delta_t = 3t^p/(c\delta)$. Now, the idea is that by the time when both terms on the right-hand side are small compared to the left-hand side (say, they are both less than half of the left-hand side), the algorithm would have stopped. Further, for large T, V_T can be upper bounded by a constant times the larger of σ^2 and $\epsilon|\mu|$ (with high probability). These two constraints then give us the required bound on T.

By applying Lemma 2 with the choice $x = \log(1/\delta_t)$, it follows that for any $t \geq 1$, with probability at least δ_t,

$$V_t \leq \sigma^2 + \sigma\sqrt{\frac{2\log(1/\delta_t)}{t}} + \frac{\log(1/\delta_t)}{3t}. \tag{3.20}$$

An application of Lemma 3 to $3\log(1/\delta_t)/t = \sigma^2$ gives that if $t \geq \frac{6}{\sigma^2}[p\log\frac{6p}{\sigma^2} + \log\frac{3}{c\delta}] = t_{\sigma^2}$ then $3\log(1/\delta_t)/t \leq \sigma^2$. Another application of Lemma 3 to $3\log(1/\delta_t)/t = \epsilon|\mu|$ gives that if $t \geq \frac{6}{\epsilon|\mu|}[p\log\frac{6p}{\epsilon|\mu|} + \log\frac{3}{c\delta}] = t_{\epsilon|\mu|}$ then $3\log(1/\delta_t)/t \leq \epsilon|\mu|$. We now define $\mathcal{E}'$ to be the event that (3.20) holds for all $t \geq 1$. Defining $\rho = \max(\sigma^2, \epsilon|\mu|)$, we get that for any $t \geq \min(t_{\sigma^2}, t_{\epsilon|\mu|})$, on $\mathcal{E}'$, $V_t \leq 3\rho$.

Thus, on $\mathcal{E} \cap \mathcal{E}'$ it holds that

$$\begin{aligned}\sqrt{\frac{2V_t\log(1/\delta_t)}{t}} + \frac{3\log(1/\delta_t)}{t} &\leq \sqrt{\frac{6\rho\log(1/\delta_t)}{t}} + \sqrt{\frac{3\rho\log(1/\delta_t)}{t}} \\ &= \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\log(1/\delta_t)}{t}}.\end{aligned} \tag{3.21}$$

Now, consider the first time $t^* \geq \min(t_{\sigma^2}, t_{\epsilon|\mu|})$ satisfying

$$\frac{\epsilon|\mu|}{1+\epsilon} \geq \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\log(1/\delta_t)}{t}}. \tag{3.22}$$

Note that t^* is non-random. Further, on $\mathcal{E} \cap \mathcal{E}'$, $t^* \geq \max(T, \min(t_{\sigma^2}, t_{\epsilon|\mu|})) = T'$. This is trivial if $T' = \min(t_{\sigma^2}, t_{\epsilon|\mu|})$. On the other hand, if $T' = T > \min(t_{\sigma^2}, t_{\epsilon|\mu|})$ then notice that Inequality (3.21) holds for time T on $\mathcal{E} \cap \mathcal{E}'$ and hence the stopping criterion (3.19) will be satisfied whenever (3.22) is satisfied. This means that the algorithm stops the latest at time t^*. Since $T' \geq T$, $t^* \geq T$ on $\mathcal{E} \cap \mathcal{E}'$.

Now, another application of Lemma 3 to Inequality (3.22) gives the bound that Inequality (3.22) is satisfied when

$$t \geq \frac{6p(1+\epsilon)^2(\sqrt{6}+\sqrt{3})^2\rho}{\epsilon^2\mu^2}\left[\frac{1}{p}\log\frac{3}{c\delta} + \log\frac{6p(1+\epsilon)^2(\sqrt{6}+\sqrt{3})^2\rho}{\epsilon^2\mu^2}\right].$$

Since the quantity on the right-hand side is at least as large as $t_{\epsilon|\mu|}$, it is an upper bound on t^*. The desired form of the bound is obtained by absorbing the additive constants into the multiplicative constant. Noticing that $\mathbb{P}[\mathcal{E} \cap \mathcal{E}'] \geq 1 - 2\delta$ finishes the proof of Part 1. □

Proof of Part 2. First, let

$$t' = C \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{1}{\epsilon|\mu|}\right)\left(\log\frac{1}{\epsilon|\mu|} + \log\frac{2}{\delta}\right). \tag{3.23}$$

where C is as defined in Part 1. Then using the definition of expectation

$$\mathbb{E}[T] = \sum_{t=1}^{\infty} t \cdot \mathbb{P}[T = t] \tag{3.24}$$

$$= \sum_{t=1}^{t'+1} t \cdot P(T = t) + \sum_{t=t'+2}^{\infty} t \cdot \mathbb{P}[T = t] \tag{3.25}$$

$$\leq 2t' + \sum_{t=t'+2}^{\infty} t \cdot \mathbb{P}[T = t], \tag{3.26}$$

where we used $\sum_{t=1}^{t'+1} t \cdot P(T = t) \leq (t'+1) \sum_{t=1}^{t'+1} P(T = t) \leq t' + 1 \leq 2t'$, where we assumed, without the loss of generality, that $t' \geq 1$.

To bound the second term in (3.26), we recall that for $t \geq t'$ whenever the confidence intervals at time t hold, the algorithm is guaranteed to stop after t samples. Hence, if the algorithm has not stopped after $t > t'$ samples, all confidence intervals between time t' and $t - 1$ must have failed. Since we can bound this probability by the probability of at least one of the confidence intervals at time $t-1$ failing, it follows that $\mathbb{P}[T = t] \leq 2d_{t-1}$. Since $d_{t-1} = c\delta(t-1)^{-p}$, it follows that

$$\mathbb{E}[T] \leq 2t' + \sum_{t=t'+2}^{\infty} t \cdot \mathbb{P}[T = t] \tag{3.27}$$

$$\leq 2t' + \sum_{t=t'+2}^{\infty} 2c'\delta(t-1)^{-p+1} \tag{3.28}$$

$$\leq 2t' + C' \tag{3.29}$$

$$\leq C''t' \tag{3.30}$$

when $p > 2$. Note that the same result can be obtained for $p > 1$, but we have chosen this argument for simplicity.

□

Proof of Part 3. Let $\mathcal{F}$ be the event that the stopping rule fails to produce an estimate with relative error ϵ,

$$\mathcal{F} = \{|\hat{\mu} - \mu| \geq \epsilon|\mu|\}$$

and let $\mathcal{E}$ be the event that the confidence intervals c_t do not fail as before (cf. (3.18)). First, we decompose the failure probability as $\mathbb{P}[\mathcal{F}] = \mathbb{P}[\mathcal{F}|\mathcal{E}]\,\mathbb{P}[\mathcal{E}] + \mathbb{P}\left[\mathcal{F}|\overline{\mathcal{E}}\right]\mathbb{P}\left[\overline{\mathcal{E}}\right]$. By construction, $\mathbb{P}\left[\overline{\mathcal{E}}\right] \leq \delta$. Then using the trivial bounds $\mathbb{P}[\mathcal{E}] \leq 1$ and $\mathbb{P}\left[\mathcal{F}|\overline{\mathcal{E}}\right] \leq 1$ we obtain $\mathbb{P}[\mathcal{F}] \leq \mathbb{P}[\mathcal{F}|\mathcal{E}] + \delta$. We now argue that $\mathbb{P}[\mathcal{F}|\mathcal{E}] = 0$.

It remains to be shown that $\hat{\mu}$ is an (ϵ, δ)-approximation of μ. So assume that the algorithm has terminated after T samples. (Notice that he algorithm stops with probability one since by Part 2, $\mathbb{E}[T] < +\infty$.) Combining the definition of event $\mathcal{E}$ with the first part of Inequality (3.16) and the definition of $\hat{\mu}$ leads to

$$\begin{aligned} |\hat{\mu}| &= 1/2 \cdot \left[(1+\epsilon) \max_{s \leq T} (|\overline{X}_s| - c_s) + (1-\epsilon) \min_{s \leq T} (|\overline{X}_s| + c_s) \right] \\ &\geq (1-\epsilon) \min_{s \leq T} \left(|\overline{X}_s| + c_s \right) \\ &\geq (1-\epsilon)|\mu| \end{aligned} \tag{3.31}$$

and

$$\begin{aligned} |\hat{\mu}| &= 1/2 \cdot \left[(1+\epsilon) \max_{s \leq T} (|\overline{X}_s| - c_s) + (1-\epsilon) \min_{s \leq T} (|\overline{X}_s| + c_s) \right] \\ &\leq (1+\epsilon) \min_{s \leq T} \left(|\overline{X}_s| - c_s \right) \\ &\leq (1+\epsilon)|\mu|. \end{aligned} \tag{3.32}$$

Inequalities (3.31) and (3.32) hold due to the fact that a confidence interval on $\overline{X}_s$ is also a confidence interval on $|\overline{X}_s|$ with equal or greater confidence. Finally, the definition of $\mathcal{E}$ and the second part of Inequality (3.16) together imply $\text{sgn}(\overline{X}_T) = \text{sgn}(\mu)$. Hence, $|\hat{\mu} - \mu| \leq \epsilon|\mu|$ and therefore, $\mathbb{P}[\mathcal{F}|\mathcal{E}] = 0$ and hence $\mathbb{P}[\mathcal{F}] = \mathbb{P}[|\hat{\mu} - \mu| \geq \epsilon|\mu|] \leq \delta$. □

3.4 Effect of Range

While our analysis of EBStop is limited to the case of X_i with range 1, extending this result to random variables with range R is straightforward.

3.4.1 The reduction approach

We begin by showing how an (ϵ, δ)-stopping rule for random variables with range 1 and a matching upper bound on its expected stopping time can be extended to random variables with range R. Let $\mathcal{S}$ be an (ϵ, δ)-stopping rule for random variables with range 1, let X be distributed with range R, and let $X' = X/R$. Now, suppose stopping rule $\mathcal{S}'$ takes X as input, runs stopping rule $\mathcal{S}$ on X' to obtain $\hat{\mu}$, and returns $R\hat{\mu}$. Then it is straightforward to show that $R\hat{\mu}$ is an (ϵ, δ)-approximation of X. Hence, $\mathcal{S}'$ is an (ϵ, δ)-stopping rule.

To see how an upper bound on the expected stopping time of $\mathcal{S}'$ can be obtained from an upper bound on the expected stopping time of $\mathcal{S}$, let $T(\epsilon, \delta, X)$ be the stopping time of $\mathcal{S}$ on X and let n be a function that satisfies

$$\mathbb{E}[T(\epsilon, \delta, X)] \leq n(\epsilon, \delta, \mu, \sigma^2).$$

Then it follows that if $T'(\epsilon, \delta, X)$ is the stopping time of S' on X, where $\mathbb{E}[X] = \mu$ and $\mathbb{V}[X] = \sigma^2$, then

$$\mathbb{E}[T'(\epsilon, \delta, X)] \leq n(\epsilon, \delta, \mu/R, \sigma^2/R^2).$$

We will refer to this method of extending a stopping rule to random variables with range R as the *reduction* approach.

3.4.2 Upper bounds

We now give upper bounds on the expected number of samples required by the EBStop and NAS algorithms required to find an (ϵ, δ)-approximation of a random variable with range R using the reduction approach. It should be noted that both algorithms can be run directly on random variables with range R, i.e. without resorting to a reduction. It should be clear that any run of any of these unmodified algorithms stops at the same time and returns the same value than running the algorithms obtained with the reduction approach described above.

Hence, it follows from Theorem 2 that EBStop can be used to find an (ϵ, δ)-approximation of a random variable with range R using an expected number of samples no greater than

$$C \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon|\mu|}\right)\left(\log\frac{R}{\epsilon|\mu|} + \log\frac{3}{\delta}\right)$$

for some universal constant C. Similarly, the NAS algorithm can be used to find and (ϵ, δ)-approximation of a random variable with range R using an expected number of samples no greater than

$$C' \cdot \frac{R^2}{\mu^2\epsilon^2} \cdot \left(\log\frac{R}{\epsilon|\mu|} + \log\frac{2}{\delta}\right)$$

for some universal constant C'.

3.4.3 Lower bound

Since the lower bound of Dagum et al. does not take the range into account we extend their result to the case of random variables distributed in $[0, R]$. The definition of universal stopping rules to random variables with range $[0, R]$ is trivial and hence the formal definition is omitted.

Theorem 3. *Let S be a universal (ϵ, δ)-stopping rule for distributions supported on $[0, R]$. Pick any $(\epsilon, \delta) \in (0, 1)^2$ and any distribution D supported on $[0, R]$ whose mean is positive. Let $N_{(\epsilon,\delta)}$ be the time when S stops on this problem with parameters (ϵ, δ). Then*

$$\mathbb{E}[N_{(\epsilon,\delta)}] \geq c \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon\mu}\right)\log\frac{2}{\delta}, \tag{3.33}$$

where μ is the mean of D and σ^2 is its variance, and $c > 0$ is a constant that is independent of $\mathcal{S}$, D, ϵ, and δ.

Note that constant c can be chosen to be the same as in Equation (1).

Proof. For a random variable X with finite variance let μ_X be its mean and σ_X^2 be its variance.

Let $\mathcal{S}$ be a universal (ϵ, δ)-stopping rule for distributions supported on $[0, R]$. Consider a stopping rule $\mathcal{S}'$ constructed from $\mathcal{S}$ as follows: $\mathcal{S}'$ works for distributions supported on $[0, 1]$. When $\mathcal{S}'$ works with Z supported on $[0, 1]$, it runs $\mathcal{S}$ on $X = ZR$ to obtain an estimate $\hat{\mu}(\epsilon, \delta, X)$. Clearly, $\hat{\mu}(\epsilon, \delta, X)/R$ is an (ϵ, δ)-approximation of μ_Z. Hence, $\mathcal{S}'$ is a universal (ϵ, δ)-stopping rule for distributions supported on $[0, 1]$. Let $N'(\epsilon, \delta, Z)$ be the number of samples consumed by $\mathcal{S}'$ on X. Then $N'(\epsilon, \delta, Z) = N(\epsilon, \delta, ZR)$ by construction, where $N(\epsilon, \delta, X)$ is the time when $\mathcal{S}$ stops when it is run with parameters ϵ, δ on *iid* copies of X.

Now, let us fix $(\epsilon, \delta) \in (0, 1)^2$ and $X \sim D$, where X is a distribution supported on $[0, R]$. Define $Z = X/R$. Hence $N(\epsilon, \delta, X) = N'(\epsilon, \delta, Z)$. By Theorem 1,

$$\mathbb{E}[N'(\epsilon, \delta, Z)] \geq c \max\left(\frac{\sigma_Z^2}{\epsilon^2 \mu_Z^2}, \frac{1}{\epsilon \mu_Z}\right) \log \frac{2}{\delta}.$$

Using $\mu_Z = \mu_X/R$ and $\sigma_Z^2 = \sigma_X^2/R^2$ we get the desired lower bound. □

Theorem 3 applies only to random variables distributed in $[0, R]$, however, it follows trivially that the same lower bound applies to stopping rules that work on random variables with range R. Hence, EBStop is at most a logarithmic term away from achieving the optimal expected stopping time.

Chapter 4

Batch Sampling

In this chapter, we present a version of the EBStop algorithm that performs batch sampling and show that with a geometric sampling schedule the $\log \frac{1}{\epsilon|\mu|}$ term in the expected number of samples taken by EBStop is reduced to $\log \log \frac{1}{\epsilon|\mu|}$. Furthermore, we introduce a new technique that allows us to test the stopping condition after each sample while maintaining the benefits of batch sampling.

4.1 Batch Sampling

The motivation behind batch sampling comes from the fact that checking the stopping criterion after each sample is wasteful when EBStop is far from stopping. To see why this is true, consider what happens when EBStop checks the stopping criterion after t samples, but cannot stop. The algorithm must construct a $1 - d_t$ confidence interval for μ, and in order to guarantee that all confidence intervals hold with probability at least δ, we require that $\sum_{t=1}^{\infty} d_t \leq \delta$. Due to this constraint, checking the stopping condition at time t reduces the mass given to d_s for $s > t$, which in turn makes the confidence intervals c_s for $s > t$ wider, and from Equation (3.17) it is clear that making the confidence intervals wider will push the stopping time back. Hence, EBStop can be made more efficient by reducing the number of times it checks the stopping criterion while it is far from stopping.

Pseudocode for a variant of EBStop that performs batch sampling is shown as Algorithm 7. The key change from EBStop is the addition of a sampling schedule in the form of a sequence of positive integers $\{t_k\}$. The sampling schedule represents the times at which the stopping condition is checked. After drawing t_k samples, Algorithm 7 constructs a $1 - d_k$ confidence interval for μ and checks the stopping criterion.

Whenever, $k \ll t_k$, it should be possible for Algorithm 7 to stop much earlier than EBStop. One possible sampling schedule, known as *arithmetic* sampling, is given by $t_k =$

Algorithm 7 EBStop with batch sampling

$t \leftarrow 1$
$l(t) \leftarrow 0$
$u(t) \leftarrow \infty$
$k \leftarrow 0$
Obtain X_1
while $(1+\epsilon)l(t) < (1-\epsilon)u(t)$ **do**
  while $t \leq t_{k+1}$ **do**
    $t \leftarrow t+1$
    Obtain X_t
  end while
  $k \leftarrow k+1$
  $c_k = \sqrt{\frac{2V_t \log(3/d_k)}{t}} + \frac{3R\log(3/d_k)}{t}$
  $l(t) \leftarrow \max(l(t-1), |\overline{X}_t| - c_k)$
  $u(t) \leftarrow \min(u(t-1), |\overline{X}_t| + c_k)$
end while
return $\operatorname{sgn}(\overline{X}_t) \cdot 1/2 \cdot [(1+\epsilon)l(t) + (1-\epsilon)u(t)]$

$m \cdot k$ for some $m > 1$. To see how such a strategy will impact the stopping time, consider checking the stopping condition after having taken t samples using arithmetic sampling. Since after t samples at most t/m confidence intervals have been constructed, the algorithm will construct a $1-d$ confidence interval where

$$d \leq \frac{c\delta}{(t/m)^p} = \frac{cm^p\delta}{t^p} = \frac{c'\delta}{t^p}.$$

Hence, an arithmetic sampling strategy only results in a change to the normalizing constant in d_t, and from Equation (3.19) it is clear that the form of the upper bound on the expected number of samples will not change.

Now, consider a *geometric* sampling schedule, where $t_k = \lceil \beta^k \rceil$ for some $\beta > 1$. Since under this schedule the stopping condition is checked at most $\log_\beta t$ times by the time t samples have been taken, it follows that

$$d \leq \frac{c\delta}{(\log_\beta t)^p}.$$

It is straightforward to show that with this value of d, an analysis of stopping times leads to equations of the form $(\log\log t)/t = c$ instead of $(\log t)/t = c$, as in the case of arithmetic sampling. While we delay the proof until Section 4.3, we will show that if T is the stopping time of Algorithm 7 employing a geometric sampling schedule, then there exists a universal constant C, such that

$$\mathbb{E}[T] \leq C \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon|\mu|}\right)\left(\log\log\frac{R}{\epsilon|\mu|} + \log\frac{1}{\delta}\right). \tag{4.1}$$

4.2 Mid-interval Stopping

While batch sampling can significantly reduce the number of samples required by EBStop, it restricts the algorithm to the sequence $\{t_k\}$ as the set of possible stopping times. When t_k grows quickly with k, batch sampling leads to many unnecessary samples being taken. For example, when employing a geometric sampling schedule with $\beta = 1.1$, it is possible that the stopping criterion could be satisfied after $\lceil \beta^k \rceil + 1$ samples, but the algorithm will not stop until it has taken $\lceil \beta^{k+1} \rceil$ samples. This can lead to as much as β times more samples than necessary being taken, and while this is only a constant, multiplicative increase, a stopping rule that is able to stop at any point is desirable.

To illustrate the efficiency of the approach we are about to propose, consider modifying a batch sampling algorithm to stop at any point through another application of the union bound. Instead of taking samples $t_k + 1, t_k + 2, \ldots, t_{k+1}$ and then checking the stopping condition with failure probability d_k, one can check the stopping condition after each sample between $t_k + 1$ and t_{k+1} with failure probability $d_k/(t_{k+1} - t_k)$. While this approach leads to earlier stopping times for very small μ and ϵ, the benefits of batch sampling become much smaller. Pseudocode for this approach is shown as Algorithm 8.

Algorithm 8 Batch EBStop with union bound anytime stopping

$t \leftarrow 1$
$l(t) \leftarrow 0$
$u(t) \leftarrow \infty$
$k \leftarrow 0$
Obtain X_1
while $(1+\epsilon)l(t) < (1-\epsilon)u(t)$ **do**
  $t \leftarrow t + 1$
  Obtain X_t
  if $t > t_{k+1}$ **then**
    $k \leftarrow k + 1$
    $d_t' \leftarrow d_k/(t_{k+1} - t_k)$
  end if
  $c_t = \sqrt{\frac{2V_t \log(3/d_t')}{t}} + \frac{3R\log(3/d_t')}{t}$
  $l(t) \leftarrow \max(l(t-1), |\overline{X}_t| - c_t)$
  $u(t) \leftarrow \min(u(t-1), |\overline{X}_t| + c_t)$
end while
return $\operatorname{sgn}(\overline{X}_t) \cdot 1/2 \cdot [(1+\epsilon)l(t) + (1-\epsilon)u(t)]$

Nevertheless, it is possible to achieve anytime stopping without the use of the union bound. The key result, due to Audibert et al., is the following variant of the empirical Bernstein bound that holds simultaneously over an interval [1]: If $t_1 \leq t_2$ for $t_1, t_2 \in \mathbb{N}$

and $\alpha \geq t_2/t_1 (\geq 1)$, then with probability at least $1 - 3e^{-x/\alpha}$, for all $t \in \{t_1, \ldots, t_2\}$ we have

$$|\overline{X}_t - \mu| \leq \sqrt{2V_t x/t} + 3x/t. \tag{4.2}$$

To apply this result to batch sampling, we first solve $1 - 3e^{-x/\alpha} = 1 - d_k$ for x, resulting in $x = \alpha \log 3/d_k$. If we then use this value of x and Equation (4.2) to construct confidence intervals for μ after each sample from $t_k + 1$ through t_{k+1}, the confidence intervals will simultaneously hold μ with probability at least $1 - d_k$. The confidence intervals can in turn be used to check the stopping condition after each sample. Algorithm 9 incorporates this idea into our batch sampling algorithm.

Algorithm 9 Batch EBStop with martingale-based anytime stopping

```
t ← 1
l(t) ← 0
u(t) ← ∞
k ← 0
Obtain X_1
while (1 + ε)l(t) < (1 − ε)u(t) do
  t ← t + 1
  Obtain X_t
  if t > t_{k+1} then
    k ← k + 1
    α ← t_{k+1}/t_k
    x ← α log 3/d_k
  end if
  c_t ← sqrt(2V_t x/t) + 3Rx/t
  l(t) ← max(l(t − 1), |X̄_t| − c_t)
  u(t) ← min(u(t − 1), |X̄_t| + c_t)
end while
return sgn(X̄_t) · 1/2 · [(1 + ε)l(t) + (1 − ε)u(t)]
```

To see how EBStop compares to Algorithm 8 and to Algorithm 9 we compare the failure probability used when evaluating the stopping criterion after $\lceil \beta^k \rceil = t$ samples by each of the algorithms, denoting this probability by f_t. The results are presented in Table 4.2. A geometric sampling schedule was assumed for the batch algorithms. Disregarding the constants, Algorithm 8 uses $f_t = 1/(t \log t)$ which, depending on the value of p, can be a very minor improvement on $f_t = 1/t^p$ as used by EBStop. Algorithm 9, roughly uses $f_t = 1/(\log t)^\beta$, which is a clear improvement over both EBStop and Algorithm 8.

EBStop	Algorithm 8	Algorithm 9
$\frac{1}{t^p}$	$\frac{1}{\lceil \log_\beta t \rceil^p} \cdot \frac{1}{t - \lceil t/\beta \rceil}$	$\left(\frac{1}{\lceil \log_\beta t \rceil}\right)^p$

Table 4.1: Failure probability used to evaluate the stopping criterion after t samples by each algorithm.

4.3 Analysis of Batch Sampling

In this section we provide an analysis of stopping times for Algorithm 9 when following a geometric sampling schedule, which we will refer to as EBGStop. We begin by proving the equivalent of Lemma 3 for the type of equations that arise under a geometric sampling schedule.

Lemma 4. *Let a, k be positive real numbers. If t' is a solution to*

$$\frac{\log(a \log t)}{t} = k \tag{4.3}$$

in terms of t, then

$$t' \leq \frac{\log\left(a \log t_0\right)}{\left[k - \frac{1}{t_0 \log t_0}\right]}, \tag{4.4}$$

where $t_0 = \max(1/k, e)$.

Proof. The proof is analogous to the proof of Lemma 3. The solution of equation (4.3) can be seen as the intersection point between a line and a $\log \log$ curve when we rewrite it as

$$\log(a \log t) = kt. \tag{4.5}$$

First, note that the slope of the line equals the slope of the tangent to the logarithmic curve at t when $t \log t = 1/k$. The solution to this equation is bounded from above by $t_0 = \max(1/k, e)$. As in the proof of Lemma 4, the intersection of the line tangent to $\log(a \log t)$ at $t \geq t_0$ with the line kt is an upper bound on t'. Since the line tangent to $\log(a \log t)$ at t_0 is given by $\log(a \log t_0) + \frac{1}{t_0 \log t_0} \cdot (t - t_0)$, solving $\log(a \log t_0) + \frac{1}{t_0 \log t_0} \cdot (t - t_0) = kt$ yields

$$t' \leq \frac{\log\left(a \log t_0\right) - \frac{1}{\log t_0}}{\left[k - \frac{1}{t_0 \log t_0}\right]} \leq \frac{\log\left(a \log t_0\right)}{\left[k - \frac{1}{t_0 \log t_0}\right]}. \tag{4.6}$$

□

Theorem 4. *Let X be a random variable distributed with range 1. Let $\mu = \mathbb{E}[X]$ and $\sigma^2 = \mathbb{V}[X]$ and assume $\mu \neq 0$. Let T be the stopping time of Algorithm 9 on X while*

following a geometric sampling schedule (i.e. $d_k = c\delta/k^p$.) Then here exists a constant C such that

$$\mathbb{P}\left[T > C \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{1}{\epsilon|\mu|}\right)\left[\log\log\frac{1}{\epsilon|\mu|} + \log\frac{3}{\delta}\right]\right] \leq 2\delta.$$

Proof. The proof is analogous to the proof of Theorem 2, but with the application of Lemma 4 in place of Lemma 3. Since Algorithm 9 differs from EBStop only in the form of c_t, following the proof of Theorem 2 until Equation (3.19), EBGStop will stop with probability at least $1-\delta$ when

$$\frac{\epsilon|\mu|}{1+\epsilon} \geq c_T. \tag{4.7}$$

Substituting c_t with the value used in Algorithm 9, Inequality (4.7) becomes

$$\frac{\epsilon|\mu|}{1+\epsilon} \geq \sqrt{\frac{2V_T\log(1/\delta_K)}{T}} + \frac{3\log(1/\delta_K)}{T}, \tag{4.8}$$

where $1/\delta_k = 3k^p/(c\delta)$ and $K = \lfloor \log_\beta T \rfloor \leq \log_\beta T$.

As before, we seek a high-probability upper bound on V_t. By applying Lemma 2 with the choice $x = \alpha\log\frac{3}{c\delta}(\log_\beta t)^p$ to V_t, it follows that for all $t \geq 1$, with probability at least $1-\delta$,

$$V_t \leq \sigma^2 + \sigma\sqrt{\frac{2\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t}} + \frac{4\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{3t}. \tag{4.9}$$

Now let $\rho = \max(\sigma^2, \epsilon|\mu|)$, then if t is a solution to

$$\frac{3\alpha p\log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}}\frac{\log t}{\log\beta}\right)}{t} = \rho, \tag{4.10}$$

then by Lemma 4 (using $3\alpha p/\rho > e$),

$$t \leq \left(1 - \frac{1}{\log\left(\frac{3\alpha p}{\rho}\right)}\right)^{-1}\frac{3\alpha p}{\rho}\left[\log\log\frac{3\alpha p}{\rho} + \frac{1}{p}\log\frac{3}{c\delta} - \log\log\beta\right] \tag{4.11}$$

$$\leq C_1\frac{3\alpha p}{\rho}\left[\log\log\frac{3\alpha p}{\rho} + \frac{1}{p}\log\frac{3}{c\delta}\right] = t_\rho. \tag{4.12}$$

We now define $\mathcal{E}'$ to be the event that (4.9) holds for all $t \geq 1$. It then follows that if the algorithm has taken at least t_ρ samples and event $\mathcal{E}'$ holds, then $V_t \leq 3\rho = 3\max(\sigma^2, \epsilon|\mu|)$.

Hence, it follows that on $\mathcal{E}$ and $\mathcal{E}'$, if $t \geq t_\rho$ then

$$\sqrt{\frac{2V_t\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t}} + \frac{3\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t} \leq \sqrt{\frac{6\rho\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t}} + \sqrt{\frac{3\rho\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t}}$$

$$\leq \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\alpha\log\frac{3}{c\delta}(\log_\beta t)^p}{t}}.$$

Now consider the smallest $t^* \geq t_\rho$ that satisfies

$$\frac{\epsilon|\mu|}{1+\epsilon} \geq \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\alpha \log \frac{3}{c\delta}(\log_\beta t)^p}{t}}.$$

As in the proof of Theorem 2, when $\mathcal{E}$ and $\mathcal{E}'$ hold, $t^* \geq T$. One final application of Lemma 4 gives

$$t^* \leq \frac{\rho(1+\epsilon)^2(\sqrt{6}+\sqrt{3})^2\alpha p}{\epsilon^2\mu^2}\left[\log\log\frac{\rho(1+\epsilon)^2(\sqrt{6}+\sqrt{3})\alpha p}{\epsilon^2\mu^2} + \frac{1}{p}\log\frac{3}{c\delta}\right].$$

Again, we can obtain the desired form of the bound by absorbing the additive constant into the multiplicative constant. Noticing that $\mathcal{E}$ and $\mathcal{E}'$ hold simultaneously with probability at least $1-2\delta$ finishes the proof. □

While we do not state them here, a bound on the expected stopping time and a proof of the (ϵ, δ)-approximation property can be obtained with arguments nearly identical to those of Theorem 2. It then follows from Theorem 4 that if T is the stopping time of EBGStop when it is used to find an (ϵ, δ)-approximation of a random variable with range R using the reduction approach of Section 3.4, then there exists a universal constant C such that

$$\mathbb{E}[T] \leq C \cdot \max\left(\frac{\sigma^2}{\epsilon^2\mu^2}, \frac{R}{\epsilon|\mu|}\right)\left(\log\log\frac{R}{\epsilon|\mu|} + \log\frac{3}{\delta}\right).$$

As with the rest of our upper bounds, the same result can be proved about EBGStop directly, but we use the reduction approach for simplicity.

Chapter 5

Experimental Results

In this chapter we explore the properties of the stopping rules we have presented through a number of simulated experiments.

5.1 Experimental Setup

In addition to EBStop and EBGStop, we evaluate $\mathcal{AA}$, NAS, and geometric NAS. In order to make the comparisons fair we set equivalent parameters in different algorithms to the same value. In the case of EBGStop and the geometric version of NAS, we set β, the factor by which both algorithms grow intervals, to 1.5 for both algorithms. Domingo et al. reported this value to work best for the NAS algorithm in their experiments [7]. Since with the exception of $\mathcal{AA}$, all of the algorithms in our comparison defined a sequence of confidence intervals $\{c_t\}$, we fixed the underlying d_t sequence to

$$d_t = \frac{\delta}{t(t+1)}$$

for all algorithms. This value is the default choice used by the NAS algorithm. Since we have found that EBGStop generally performs better for other settings of these parameters, we also include results for EBGStop with our default choices $d_k = c/k^p$, $p = 1.1$, and $\beta = 1.1$. We denote EBGStop with these parameter choice by EBGStop*.

5.2 Effect of Variance

The primary reason for developing EBStop was the need for an algorithm that is able to take advantage of variance like $\mathcal{AA}$ without the restriction to nonnegative random variables. In this section we compare how well the various stopping rules are able to exploit variance.

Let $U(a, b, m)$ denote the average of m Uniform(a, b) random variables. Then the expected value and variance of $U(a, b, m)$ are $(a + b)/2$ and $(b - a)^2/(12m)$ respectively.

Since the aim of this experiment is to study the effect of the variance on stopping times, we fix a to 0 and b to 1, and vary m to obtain a number of random variables with a fixed mean but different variances. We ran each stopping rule 100 times on $U(0, 1, m)$ random variables for $m = 1, 5, 10, 50, 100, 1000$, $\epsilon = 0.01$ and $\delta = 0.1$. Figure 5.1 shows the average number of samples taken by each algorithm for each value of m. Logarithmic scale was used on the y-axis for clarity.

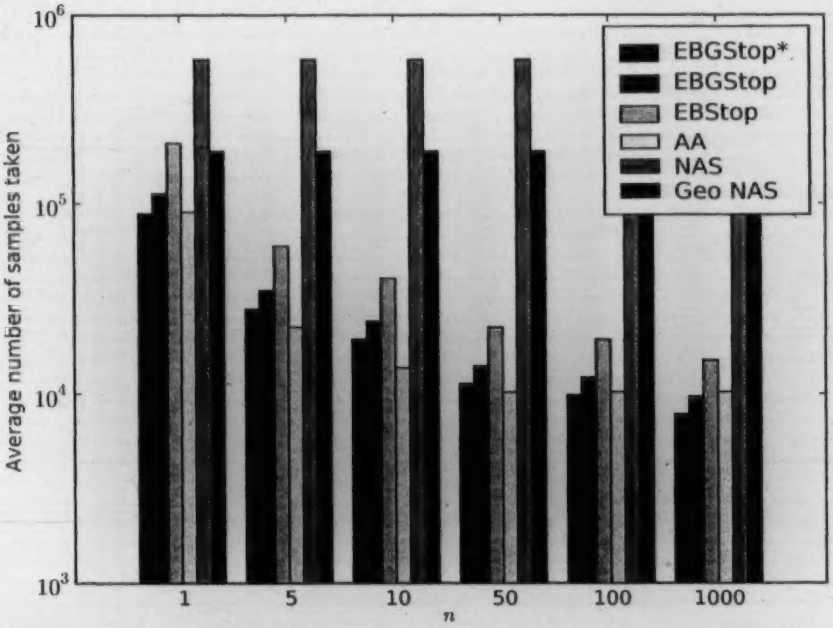

Figure 5.1: Average number of samples required to find $(0.01, 0.1)$-approximations of $U(0, 1, m)$ random variables for $m = 1, 5, 10, 50, 100, 1000$. The results are averaged over 100 runs.

Figure 5.1 suggests that variance has no effect on the expected stopping time of NAS and geometric NAS algorithms. This is not surprising as the sample variance does not appear in the stopping condition for either algorithm.

Unlike the NAS algorithms, all variants of EBStop along with the $\mathcal{AA}$ algorithm are able to take advantage of variance information, but the exact effect of the variance differs between $\mathcal{AA}$ and EBStop. The behaviour of the $\mathcal{AA}$ algorithm seems to fall into two modes. For $m = 1, 5, 10$, the algorithm requires fewer samples as the variance decreases with larger

m. However, for $m = 50, 100, 100$, the $\mathcal{AA}$ algorithm seems to require the same number of samples for all three random variables, even though the variance of $U(0, 1, 1000)$ is 20 times smaller than the variance $U(0, 1, 50)$. On the other hand, all variants of EBStop require fewer and fewer samples as the variance decreases, at least for the distributions that we tested EBStop on.

However, the theory predicts that both algorithms have two modes: When σ^2 decreases and it is above $c\epsilon\mu$ for some constant c, then the number of samples decreases with σ^2. However, when σ^2 decreases below $c\epsilon\mu$, no further decrease of the required number of samples will be experienced. Here c is a constant that depends on the algorithm. For $\mathcal{AA}$ it seems that this constant is fairly large, while for EBStop and its variants it is much smaller. (The fact that in the bounds σ^2 is compared directly to $\epsilon\mu$, i.e., that the bound depends on $\max(\sigma^2, \epsilon\mu)$ instead of $\max(\sigma^2, c\epsilon\mu)$ is a side-effect of the way the analysis is done.)

In the case of the $\mathcal{AA}$ algorithm, the constant's value is determined by the desired accuracy of the presampling step. In the case of EBStop and its variants the constant c is determined by how the two terms in the empirical Bernstein bound interact with each other. In order to understand this, recall that these algorithms can be expected to stop when

$$1 \gtrapprox \sqrt{\left(\frac{2\sigma^2}{\epsilon^2\mu^2}\right)\frac{\log(1/\delta_t)}{t}} + \left(\frac{3R}{\epsilon\mu}\right)\frac{\log(1/\delta_t)}{t} \tag{5.1}$$

(cf. Equation (3.19)), assuming that this stopping time t^* is large enough so that $V_{t^*} \approx \sigma^2$. When $\sigma \to 0$ the second term becomes dominating and the dependence of t^* on σ will be negligible. In particular, $\log(1/\delta_t)/t \lessapprox \epsilon\mu/(3R)$ must be satisfied before Inequality (5.1) will be satisfied and this puts a lower bound on t^*. This lower bound is independent of σ. Further, for small values of σ the actual cutoff point will be arbitrarily close to this lower bound. Hence, lowering the value of σ does not change lead to an improvement in the performance. This mode of behaviour is seen in Figure 5.1 for $m = 50, 100, 1000$. Even though the differences in variance for these values of m are huge, the stopping times are similar. However, when σ has a large enough value, the first term dominates. This mode of behaviour can be seen in Figure 5.1 for $m = 1, 5, 10$. While the differences in variances are much smaller than between $m = 50, 100, 1000$, when $m = 10$, almost an order of magnitude fewer samples are required to stop than when $m = 1$.

5.3 General Efficiency

5.3.1 Low Variance

We again test the stopping rules on $U(a, b, m)$ random variables. However, instead of keeping the mean fixed and studying the effect of the variance, we fix the variance and vary the mean. We fix m at 10 and vary a and b to obtain the values $\mu = 0.9, 0.7, 0.5, 0.3, 0.1$ while $b - a$ is 0.2. We used the values $\epsilon = 0.1$ and $\delta = 0.1$ in this set of experiments. The variance is small enough that EBStop, its variants, and $\mathcal{AA}$ should take a number of samples in the order of $1/(\epsilon\mu)$. We also expect both variants of the NAS algorithm to take a number of samples on the order of $1/\epsilon^2\mu^2$. Figure 5.2 shows the average number of samples taken by each algorithm for each value of μ. We again use logarithmic scale on the y-axis for clarity.

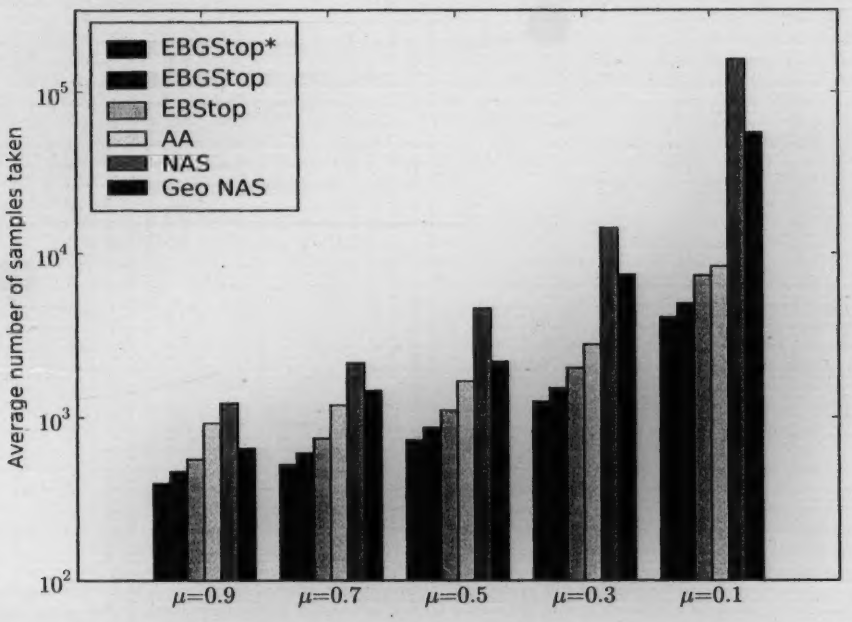

Figure 5.2: Average number of samples required to find $(0.1, 0.1)$-approximations of $U(a, b, 10)$ random variables with varying means. The results are averaged over 100 runs.

Figure 5.2 shows that both variants of the NAS algorithm quickly fall behind the other algorithms as μ decreases. It seems that, as the theory suggests, the $\mathcal{AA}$ algorithm and

all variants of EBStop require $1/\mu$ times fewer samples than NAS. While the comparison has been done on nonnegative random variables in order to include $\mathcal{AA}$, it should be emphasized that on signed random variables EBStop can be drastically more efficient than the NAS algorithm.

5.3.2 High Variance

While the previous experiment showed that both EBStop and the $\mathcal{AA}$ algorithm can exploit low-variance situations to require on the order of $1/\epsilon|\mu|$ samples to stop, how well do they perform when the variance is large? To examine this scenario, we include a comparison on Bernoulli random variables. Since Bernoulli random variables have maximal variance of all bounded random variables, the advantage of variance estimation should be diminished. Nevertheless, if μ and σ^2 are the mean and variance of a Bernoulli random variable, then $\sigma^2 = \mu(1-\mu)$. Hence, when μ is small, EBStop and $\mathcal{AA}$ should require on the order of

$$\max\left(\frac{\mu(1-\mu)}{\epsilon^2\mu^2}, \frac{1}{\epsilon\mu}\right) = \max\left(\frac{1-\mu}{\epsilon^2\mu}, \frac{1}{\epsilon\mu}\right) \approx \frac{1}{\epsilon^2\mu} \tag{5.2}$$

samples to stop.

Figure 5.3 shows the average number of samples required by each algorithm to find a $(0.1, 0.1)$-approximation of a number of Bernoulli random variables. As predicted by Equation (5.2), when μ is small, $\mathcal{AA}$ and all variants of EBStop seem to require $1/\mu$ times fewer samples than NAS. Somewhat surprisingly, the geometric version of NAS required fewer samples than even the tuned version of EBGStop for $\mu = 0.9$ and $\mu = 0.5$, but not for $\mu = 0.99$. This is likely happening because for intermediate values of μ, such as 0.9 and 0.5, the square root and the linear terms in the empirical Bernstein bound are of approximately equal magnitude when EBStop is close to stopping. This has the effect of roughly doubling the magnitude of the constants associated with the bound and slightly increasing the required number of samples.

It is also interesting to note that all variants of EBStop outperformed the $\mathcal{AA}$ algorithm in both experiments where we varied the mean, even though the bounds on the expected number of samples taken by EBStop possess an extra logarithmic term. This term grows without bound as ϵ or μ approach 0, hence, on nonnegative random variables, the $\mathcal{AA}$ can be expected to outperform EBStop when this is the case. However, we have not seen this in our experiments.

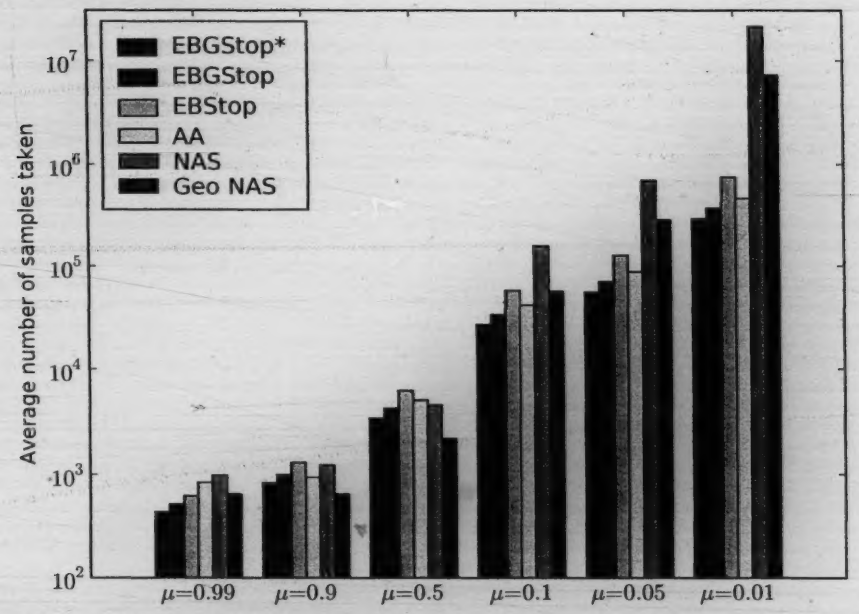

Figure 5.3: Average number of samples required to find $(0.1, 0.1)$-approximations of Bernoulli random variables with varying means. The results are averaged over 100 runs.

5.4 Coverage

In Chapter 2, we estimated the coverage of a stopping rule that uses the Central Limit Theorem in order to determine whether it is smaller or lager than $1 - \delta$. While all of the stopping rules we evaluated in this chapter guarantee that their coverage is at least $1 - \delta$, we calculated the sample coverage achieved by the stopping rules evaluated in this chapter on each of the three experiments described above. Perhaps somewhat surprisingly, there was not a single occurrence of a stopping rule returning an estimate with relative error greater than ϵ. Since each stopping rule was run 100 times on 17 different random variables, this suggests that these stopping rules are extremely conservative.

Depending on the motivation behind using a stopping rule, the overly conservative nature of such stopping rules can be seen as both an advantage and a disadvantage. If it is important to guarantee that the approximations are within ϵ relative error with probability at least $1 - \delta$, the stopping rules in this chapter are a good choice. If, on the other hand, one

is willing to tolerate coverage smaller than $1-\delta$, stopping rules based on asymptotic results will be much more efficient, though it is clear that efficiency alone cannot be the goal.

Chapter 6

Absolute Error

In this chapter we consider the simpler but related problem of estimating the mean of a random variable up to a given absolute error. As before, let $X_1, X_2, X_3, \ldots$ be *iid* random variables with mean μ. If a stopping rule returns an estimate $\hat{\mu}$ that satisfies

$$\mathbb{P}\left[|\hat{\mu} - \mu| \leq \epsilon\right] \geq 1 - \delta, \tag{6.1}$$

then we refer to $\hat{\mu}$ as an *absolute* (ϵ, δ)-approximation. We use the techniques used in the development of EBStop to derive a novel algorithm for finding absolute (ϵ, δ)-approximations and provide both a theoretical and an empirical analysis of its properties.

6.1 Non-adaptive approach

The problem of finding an absolute (ϵ, δ)-approximation is inherently easier than the problem of finding an (ϵ, δ)-approximation. Since the number of samples necessary to find an (ϵ, δ)-approximation depends on the mean μ, any stopping rule that finds an (ϵ, δ)-approximation must be *adaptive* in the sense that its stopping condition must depend on the samples. On the other hand, stopping rules for finding absolute (ϵ, δ)-approximations do not have to make use of the samples in the stopping condition.

To give an example of a non-adaptive approach, we recall that if X_i are bounded with range R, then from Hoeffding's inequality

$$\mathbb{P}\left[\left|\overline{X}_n - \mu\right| > \epsilon\right] \leq 2e^{-2n\epsilon^2/R^2}. \tag{6.2}$$

By solving for the smallest n for which the right-hand side of Inequality (6.2) is greater than δ we get that if

$$n > \frac{R^2 \log \frac{2}{\delta}}{2\epsilon^2} = n_{\epsilon}, \tag{6.3}$$

then $\mathbb{P}\left[\left|\overline{X}_n - \mu\right| > \epsilon\right] \leq \delta$. Hence, it is enough to take the average of n_ϵ samples in order to find an absolute (ϵ, δ)-approximation.

While this simple, non-adaptive approach works, it is not difficult to see that it can be improved upon by an adaptive one that makes use of variance information. In particular, it should be possible to reduce the dependence of the number of samples on R^2 to a dependence on R and σ^2. This is indeed our goal here.

6.2 Empirical Bernstein Stopping for Absolute Error

In this section we use the methods developed in Chapters 3 and 4 to obtain an efficient stopping rule for finding absolute (ϵ, δ)-approximations of bounded random variables.

6.2.1 The Algorithm

Following the development of EBStop, we rely on a sequence $\{c_t\}$ such that the event

$$\mathcal{E} = \{\left|\overline{X}_t - \mu\right| \leq c_t, \forall t \in \mathbb{N}^+\}$$

occurs with probability at least $1 - \delta$. In particular, we make the choice of using c_t based on batch sampling with a geometric sampling schedule as defined in Section 4.2. Having defined c_t, it is trivial to construct a stopping rule for finding absolute (ϵ, δ)-approximations. One can simply stop as soon as $c_t \leq \epsilon$ and return $\overline{X}_t$ as the estimate. We will refer to this algorithm as EBAStop and present pseudocode for it as Algorithm 10.

Algorithm 10 Algorithm EBAStop.

```
c_1 ← ∞
t ← 1
k ← 0
Obtain X_1
while c_t > ε do
  t ← t + 1
  Obtain X_t
  if t > t_{k+1} then
    k ← k + 1
    α ← t_k / t_{k+1}
    x ← α log 3/d_k
  end if
  c_t ← sqrt(2 V_t x / t) + 3Rx/t
end while
return X̄_t
```

We need to show that EBAStop terminates with probability 1 and returns an absolute (ϵ, δ)-approximation upon termination. The verify the first property, we recall that

$\lim_{t\to\infty} \tilde{c}_t = 0$. Since the algorithm terminates when $c_t \leq \epsilon$, we see that the stopping condition will be satisfied for large-enough t. Now suppose that the stopping condition is satisfied and event $\mathcal{E}$ holds. Then $\left|\overline{X}_t - \mu\right| \leq c_t$ and $c_t \leq \epsilon$, hence $\overline{X}_t$ is an absolute (ϵ, δ)-approximation of μ.

6.2.2 Analysis

As we have done with the other algorithms we have proposed, we derive a high-probability upper bound on the stopping time of EBAStop.

Theorem 5. *Let X be a random variable distributed with range 1. Let $\mu = \mathbb{E}[X]$ and $\sigma^2 = \mathbb{V}[X]$ and assume $\mu > 0$. Let T be the stopping time of EBAStop on X while following a geometric sampling schedule. Then here exists a constant C such that*

$$\mathbb{P}\left[T > C \cdot \max\left(\frac{\sigma^2}{\epsilon^2}, \frac{1}{\epsilon}\right)\left[\log\log\frac{1}{\epsilon} + \log\frac{3}{\delta}\right]\right] \leq 2\delta.$$

Proof. The proof is analogous to the proof of Theorem 4. EBAStop stops when $c_T \leq \epsilon$, or if we substitute the full expression for c_t, when

$$\sqrt{\frac{2\alpha V_T \log(1/\delta_K)}{T}} + \frac{3\alpha \log(1/\delta_K)}{T} \leq \epsilon, \tag{6.4}$$

where $1/\delta_k = 3k^p/(c\delta)$ and $K = \lfloor \log_\beta T \rfloor \leq \log_\beta T$. Now, as we have done in the proof of Theorem 4, we seek a a high-probability upper bound on V_t. By applying Lemma 2 with the choice $x = \alpha \log \frac{3}{c\delta}(\log_\beta t)^p$ to V_t we obtain that for all $t \geq 1$, with probability at least $1 - \delta$,

$$V_t \leq \sigma^2 + \sqrt{\sigma^2}\sqrt{\frac{2\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t}} + \left[\sqrt{\frac{4\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{3t}}\right]^2. \tag{6.5}$$

Let $\rho = \max(\sigma^2, \epsilon)$, then if t is a solution to

$$\frac{3\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t} = \rho, \tag{6.6}$$

by Lemma 4

$$t \leq \frac{3\alpha p}{\rho}\left[\log\log\frac{3\alpha p}{\rho} + \frac{1}{p}\log\frac{3}{c\delta}\right] = t_\rho. \tag{6.7}$$

We now define $\mathcal{E}'$ to be the event that (6.8) holds for all $t \geq 1$. It then follows that if EBAStop has taken at least t_ρ samples and event $\mathcal{E}'$ holds, then $V_t \leq 3\rho = 3\max(\sigma^2, \epsilon)$.

It then follows that when $\mathcal{E}$ and $\mathcal{E}'$ hold, and $t \geq t_\rho$, then

$$\sqrt{\frac{2\alpha V_t p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t}} + \frac{3\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t} \tag{6.8}$$

$$\leq \sqrt{\frac{6\rho\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t}} + \sqrt{\frac{3\rho\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t}} \tag{6.9}$$

$$\leq \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\alpha p \log\left(\left(\frac{3}{c\delta}\right)^{\frac{1}{p}} \frac{\log t}{\log\beta}\right)}{t}}. \tag{6.10}$$

Now consider the smallest $t^* \geq t_\rho$ that satisfies

$$\epsilon \geq \left(\sqrt{6}+\sqrt{3}\right)\sqrt{\frac{\rho\alpha \log\frac{3}{c\delta}(\log_\beta t)^p}{t}}.$$

As in the proof of Theorem 2, when $\mathcal{E}$ and $\mathcal{E}'$ hold, $t^* \geq T$. We can apply Lemma 4 one more time to obtain

$$t^* \leq \frac{\rho(\sqrt{6}+\sqrt{3})^2\alpha p}{\epsilon^2}\left[\log\log\frac{\rho(\sqrt{6}+\sqrt{3})^2\alpha p}{\epsilon^2} + \frac{1}{p}\log\frac{3}{c\delta}\right]. \tag{6.11}$$

The desired form of the bound can be obtained by absorbing the additive constant into the multiplicative constant when ϵ and δ are small. Finally, noticing that $\mathcal{E}$ and $\mathcal{E}'$ hold simultaneously with probability at least $1-2\delta$ finishes the proof. □

We can then use Theorem 5 and part b of Theorem 2 to obtain that there exists a universal constant C such that

$$\mathbb{E}[T] \leq C \cdot \max\left(\frac{\sigma^2}{\epsilon^2}, \frac{1}{\epsilon}\right)\left[\log\log\frac{1}{\epsilon} + \log\frac{3}{c\delta}\right].$$

Hence, if we disregard the logarithmic terms, the adaptive approach used in EBAStop requires on the order of $\max\left(\frac{\sigma^2}{\epsilon^2}, \frac{1}{\epsilon}\right)$ samples, while the non-adaptive approach requires on the order of $\frac{1}{\epsilon^2}$ samples. This implies that when the variance is small, the adaptive approach should be able to stop substantially earlier.

Nevertheless, the $\log\log\frac{1}{\epsilon}$ term can be made arbitrarily large by using a sufficiently small value of ϵ. We can get a general idea of how small ϵ has to be for this term to become non-negligible by considering the case of a Bernoulli random variable with mean μ. This random variable has variance $\mu(1-\mu)$ and it is the largest variance achievable by a random variable with mean μ and range 1. If we consider the logarithmic term to be non-negligible when

$$\frac{\sigma^2}{\epsilon^2}\log\log\frac{1}{\epsilon} > \frac{1}{\epsilon^2},$$

or equivalently when

$$\log \log \frac{1}{\epsilon} > \frac{1}{\sigma^2}, \tag{6.12}$$

we can solve for the smallest ϵ for which this is true. In the Bernoulli case, Inequality (6.12) is satisfied when

$$\epsilon < \sqrt{\frac{1}{\exp\left(\exp\left(\frac{1}{\mu(1-\mu)}\right)\right)}}. \tag{6.13}$$

By plugging in values of μ into Inequality (6.13) we get that the logarithmic term becomes non-negligible when $\epsilon < 10^{-12}$ for $\mu = 0.5$ and when $\epsilon < 10^{-113}$ for $\mu = 0.2$. Hence, ϵ would have to be really small for the logarithmic term to be sufficiently large.

As in the case of relative error, we can use the reduction approach of Section 3.4 to obtain an upper bound on the expected stopping time of EBAStop when used on random variables with range R. It is easy to show that if T is the stopping time of EBAStop in this case, then

$$\mathbb{E}[T] \leq C \cdot \max\left(\frac{\sigma^2}{\epsilon^2}, \frac{R}{\epsilon}\right)\left[\log \log \frac{R}{\epsilon} + \log \frac{3}{\delta}\right]$$

for some universal constant C.

6.2.3 Mixture of Stopping Rules

Based on our analysis it is clear that the Hoeffding-based stopping rule and our adaptive approach each have their own merits. When the variance is small compared to ϵ, the adaptive approach should only require on the order of $\frac{1}{\epsilon}$ samples. On the other hand, when ϵ is really small, the Hoeffding-based approach should be able to stop earlier than EBAstop because the log log term in the bound on the expected stopping time of EBAStop will be large.

How can we decide which algorithm to use in practice? Instead of trying to decide which stopping rule is likely to stop first when faced with a particular scenario we can combine both stopping rules into a single stopping rule. Let $T_{Hoeff}(\delta)$, and $T_{EB}(\delta)$ be the number of samples required to find an absolute (ϵ, δ)-approximation of a random variable X by the Hoeffding-based and adaptive methods respectively. The mixture stopping rule stops after $\min(T_{Hoeff}(\delta/2), T_{EB}(\delta/2))$ samples. The stopping time of this rule should be a constant worse than $\min(T_{Hoeff}(\delta), T_{EB}(\delta))$.

6.3 Experimental Results

Theorem 5 suggests that our adaptive approach should require significantly fewer samples than the non-adaptive approach when the variance is small and that the two approaches

should perform comparably when the variance is large. We compared the average number of samples required by each method to find an absolute (ϵ, δ)-approximations of random variables with a fixed mean but different variances. We ran each stopping rule 100 times on $U(0, 1, m)$ random variables for $m = 1, 5, 10, 50, 100, 1000$, $\epsilon = 0.01$ and $\delta = 0.01$. Figure 6.1 shows the average number of samples taken by each algorithm for each value of m. *Hoeffding* denotes the non-adaptive approach, *EBAStop* denotes our adaptive approach, while *Mixture* denotes the combination of the two approaches. Logarithmic scale was used on the y-axis for clarity.

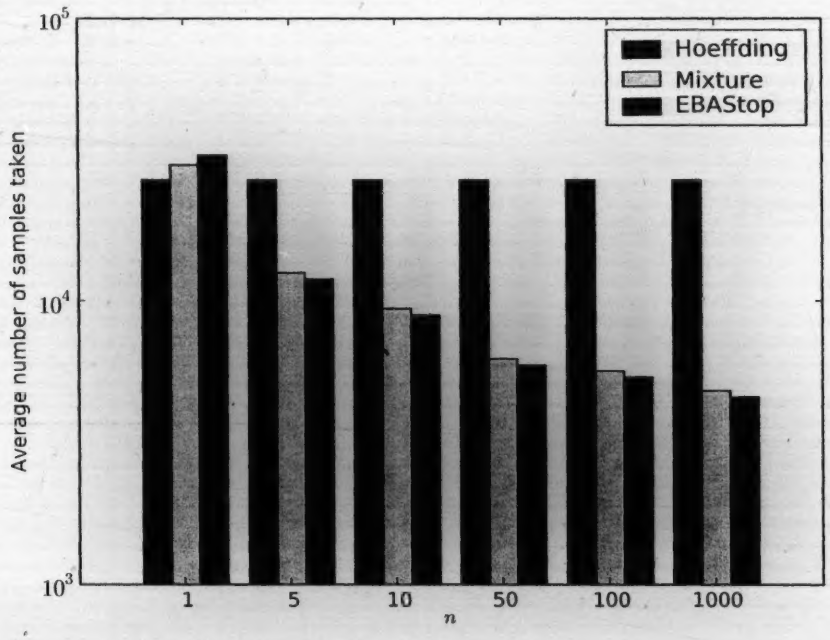

Figure 6.1: Comparison of absolute (ϵ, δ)-stopping rules on averages of m Uniform(0,1) random variables for $m = 1, 5, 10, 50, 100, 1000$.

As expected, the adaptive approach requires fewer samples to stop as the variance decreases. The variance of $U(0, 1, 1)$ is the largest of the 6 random variables, and the Hoeffding-based approach actually manages to stop earlier than the adaptive approach. However, the adaptive approach stops much earlier on $U(0, 1, 5)$ and $U(0, 1, 10)$. The reduction in the stopping times is much smaller for $m = 50, 100, 1000$, but this is not surprising because for these random variables our algorithm should require on the order of $1/\epsilon$

samples, reducing the benefit of variance estimation. The mixture of the two stopping rules performs almost as well as the better of the two rules in all cases.

In the second experiment, we compared the stopping times of the stopping rules when finding an absolute (ϵ, δ)-approximation of a $U(0, 1, 3)$ random variable for different values of ϵ. The results are resented in Figure 6.2. For large values of ϵ, the Hoeffding-based approach stops much earlier than our adaptive rule because the overhead of being adaptive is too high when a small number of samples is sufficient. When ϵ is small, our adaptive rule stops earlier by exploiting the small variance of the $U(0, 1, 3)$ random variable. As expected, in both cases, the mixture of the two stopping rules requires only a few more samples than the best stopping rule.

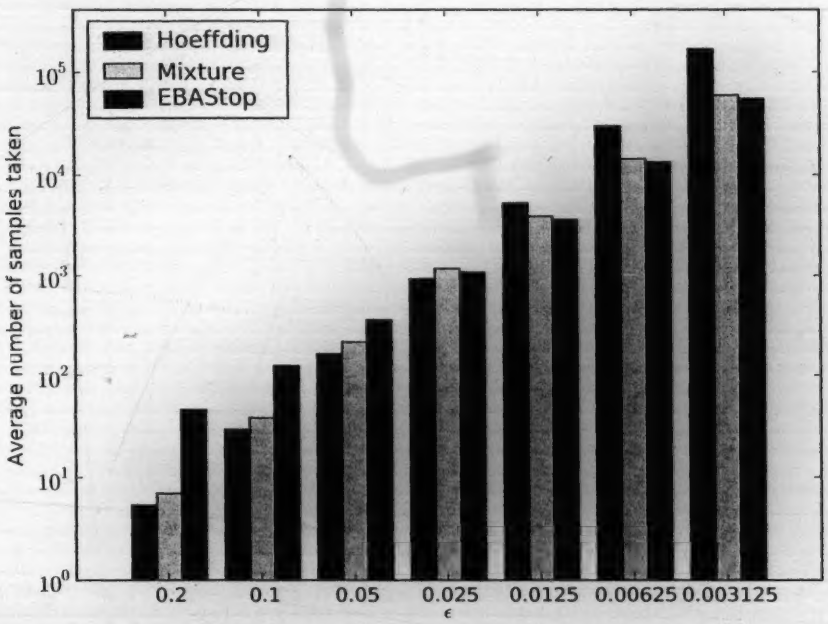

Figure 6.2: Comparison of absolute (ϵ, δ)-stopping rules on a $U(0, 1, 3)$ random variable for different values of ϵ.

6.4 Conclusions

We have presented an adaptive algorithm for finding absolute (ϵ, δ)-approximations of bounded random variables. While the algorithm is able to stop much earlier than a non-

adaptive approach when the variance is small, its expected stopping time as ϵ goes to 0 becomes larger than that of the non-adaptive approach. We showed that a mixture of the two approaches may be a good alternative to the Hoeffding-based approach in practice as it will never stop much later than the Hoeffding-based stopping rule but can, in some settings, stop much earlier.

Chapter 7

Conclusion

7.1 Summary of Contributions

The main contribution of this thesis is the introduction of the EBStop algorithm - a near-optimal stopping rule for finding (ϵ, δ)-approximations of bounded random variables. The key advantage over previous approaches is the use of empirical Bernstein bounds, which allow our algorithm to stop much earlier than its competitors when the variance is small. We also show how a version of the empirical Bernstein bound that holds over an interval can be used to make our algorithm much more efficient by grouping deviation bounds. The resulting algorithm achieves a better bound on expected stopping time and performs well in practice.

Finally, we applied our techniques to obtain a novel algorithm for finding absolute (ϵ, δ)-approximations. While our new algorithm required much fewer samples than the standard approach based on Hoeffding's inequality when the variance is small, it performed poorly in other settings. We then showed that a combination of these two approaches into a mixture stopping rule yields an algorithm that performs almost as well as the better of the two approaches in all situations.

7.2 Future Work

While EBGStop is currently the most efficient stopping rule for finding (ϵ, δ)-approximations of bounded random variables there is considerable room for improvement.

The first interesting question is whether the lower bound due to Dagum et al. is achievable in the case of bounded random variables. EBGStop comes to within a log log term involving ϵ and $|\mu|$ of achieving this lower bound. This term is the result of applying a union bound over time, and one possibility for eliminating it is by assigning the failure

probability given to the confidence interval after t samples adaptively. Since knowing μ and σ^2 in advance would allow us to determine the optimal stopping time, the hope is that using estimates of μ and σ^2 to adaptively construct the sequence of confidence intervals would allow us to come close to achieving the optimal stopping time.

Another important direction for future research is improving the coverage of nonparametric stopping rules. As we have already noted in Section 2.3, all of the existing (ϵ, δ)-stopping rules are extremely conservative and come nowhere near achieving coverage of $1 - \delta$. Some of the inefficiency stems from the use of the union bound. When EBStop is close to stopping, the confidence interval used in evaluating the stopping condition is much more conservative than $1 - \delta$. Constructing the sequence of failure probabilities $\{d_t\}$ adaptively or avoiding the use of the union bound over time all together are two promising approaches. Further improvements of the coverage could be obtained by developing better bounds to be used in place of the empirical Bernstein bound.

Chapter 8

Appendix

8.1 Probability Inequalities

Inequalities that bound the probability that a sample mean will deviate from its expected value by more than some value ϵ are an important tool for developing efficient stopping algorithms. This appendix reviews the two bounds used in this thesis.

8.1.1 Hoeffding's Inequality

Let $X_1, \ldots, X_t$ real-valued *i.i.d.* random variables with range R and, mean μ, and let $\overline{X}_t = 1/t \sum_{i=1}^{t} X_i$. Hoeffding's inequality [10] states that for any $\epsilon > 0$

$$\mathbb{P}\left[\overline{X}_t - \mu > \epsilon\right] \leq e^{-2t\epsilon^2/R^2}. \tag{8.1}$$

One can use Hoeffding's inequality to obtain that for any $\delta \in (0, 1)$, with probability at least $1 - \delta$

$$|\overline{X}_t - \mu| \leq R\sqrt{\frac{\log(2/\delta)}{2t}}. \tag{8.2}$$

8.1.2 Empirical Bernstein Bounds

The *empirical Bernstein bound* [2] states that with probability at least $1 - \delta$

$$|\overline{X}_t - \mu| \leq \sqrt{\frac{2V_t \log(3/\delta)}{t}} + \frac{3R \log(3/\delta)}{t}, \tag{8.3}$$

where V_t is the empirical variance of $X_1, \ldots, X_t$: $V_t = \frac{1}{t}\sum_{i=1}^{t}(X_i - \overline{X}_t)^2$. Note that the square root term in Inequality (8.3) is very similar to square root term in Hoeffding's bound, except in that the empirical standard deviation appears in Inequality (8.3) instead of the range R. The additional linear term appearing in the empirical Bernstein bound is necessary because the empirical standard deviation can be 0.

Bibliography

[1] J.-Y. Audibert, R. Munos, and Cs. Szepesvári. Variance estimates and exploration function in multi-armed bandit. Technical Report 07-31, Certis - Ecole des Ponts, 2007. http://cermics.enpc.fr/~audibert/RR0731.pdf.

[2] Jean Yves Audibert, Remi Munos, and Csaba Szepesvári. Tuning bandit algorithms in stochastic environments. In *18th International Conference on Algorithmic Learning Theory*, pages 150–165, 2007.

[3] Joseph K. Bradley and Robert Schapire. Filterboost: Regression and classification on large datasets. In *Advances in Neural Information Processing Systems 20*, pages 185–192, 2008.

[4] Paul Dagum, Richard Karp, Michael Luby, and Sheldon Ross. An optimal algorithm for Monte Carlo estimation. *SIAM Journal on Computing*, 29(5):1484–1496, 2000.

[5] Anirban DasGupta. *Asymptotic Theory of Statistics and Probability*. Springer, 2008.

[6] Carlos Domingo, Ricard Gavalda, and Osamu Watanabe. Adaptive sampling methods for scaling up knowledge discovery algorithms. In *Discovery Science*, pages 172–183, 1999.

[7] Carlos Domingo and Osamu Watanabe. Scaling up a boosting-based learner via adaptive sampling. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 317–328, 2000.

[8] Pedro Domingos and Geoff Hulten. Learning from infinite data in finite time. In *Advances in Neural Information Processing Systems 14*, pages 673–680, 2002.

[9] Zdenek Hlavka. *Robust Sequential Methods*. PhD thesis, Charles University, Department of Probability and Mathematical Statistics, March 2000.

[10] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[11] Michael Holmes, Alexander Gray, and Charles Isbell. Ultrafast Monte Carlo for statistical summations. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 673–680. MIT Press, Cambridge, MA, 2008.

[12] Volodymyr Mnih, Csaba Szepesvári, and Jean-Yves Audibert. Empirical Bernstein stopping. In Andrew McCallum and Sam Roweis, editors, *Proceedings of the 25th Annual International Conference on Machine Learning (ICML 2008)*, pages 672–679. Omnipress, 2008.

[13] Nitis Mukhopadhyay. *Probability and Statistical Inference*. Marcel Dekker, March 2000.

[14] David Siegmund. *Sequential Analysis: Tests and Confidence Intervals*. Springer, August 1985.

[15] Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference (Springer Texts in Statistics)*. Springer, September 2004.